Fig.1

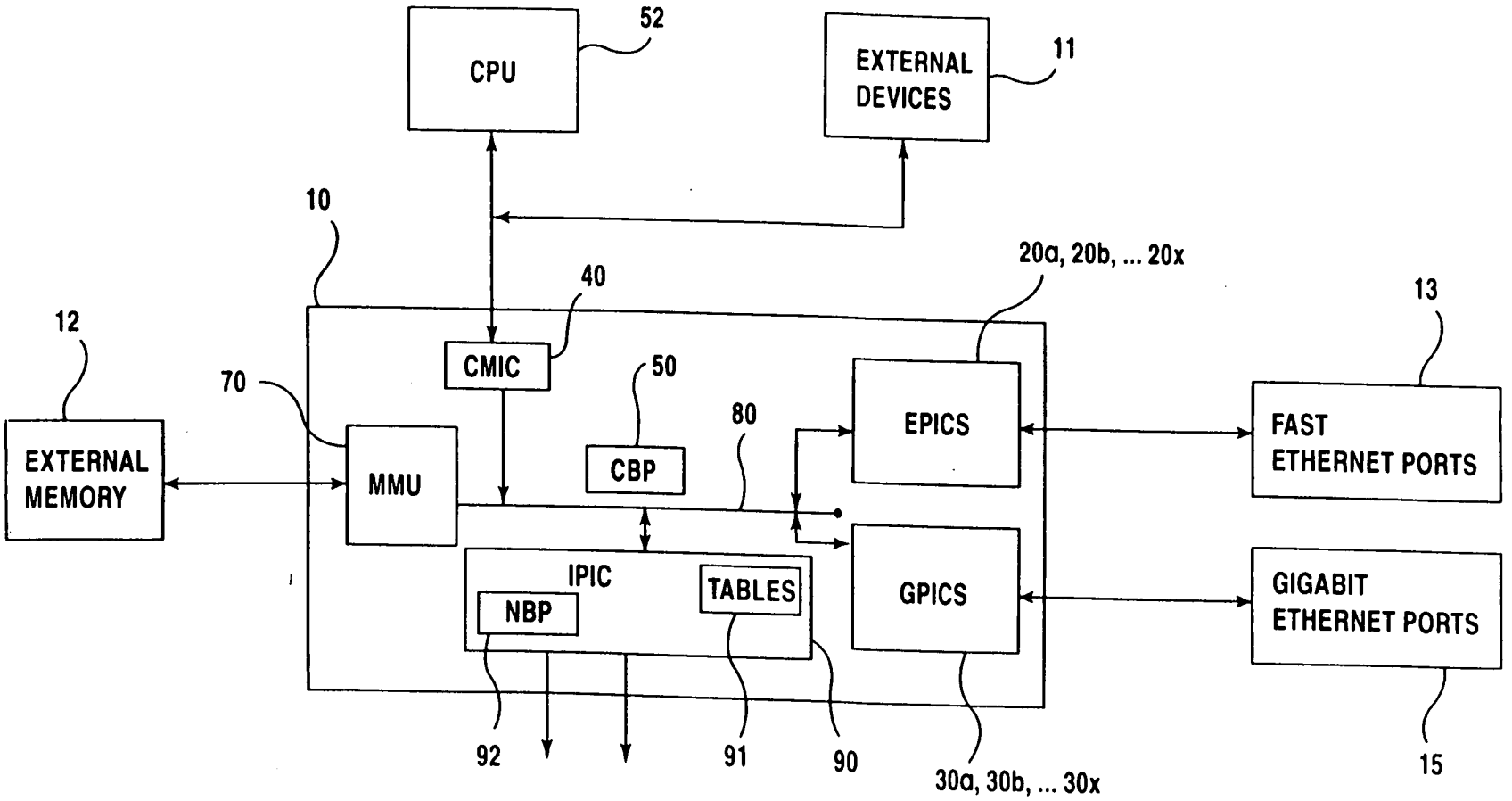


Fig.2

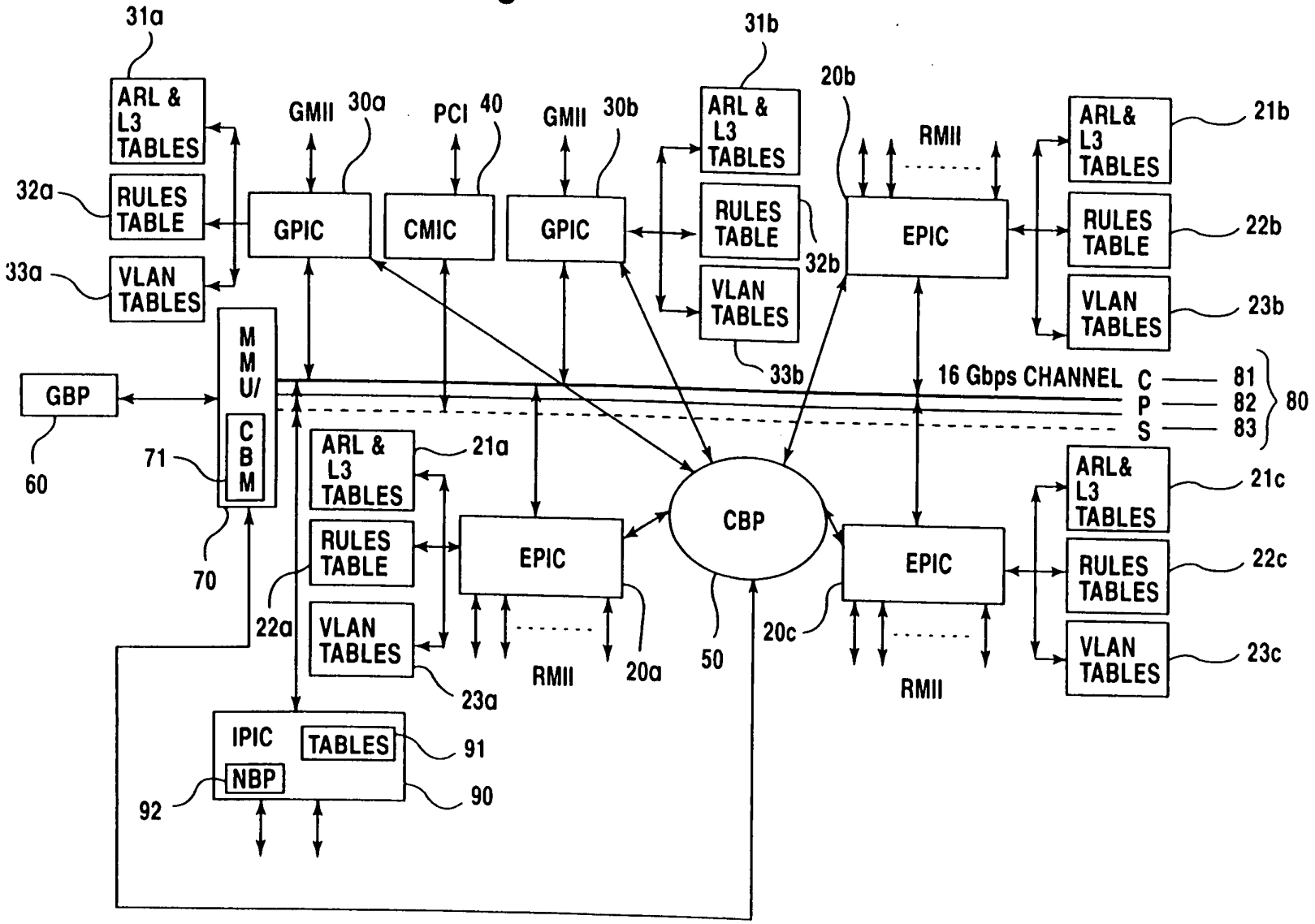

31a
ARL & L3 TABLES
32a
RULES TABLE
33a
VLAN TABLES
GMII
30a
PCI
40
GMII
30b
GPIC
CMIC
GPIC
31b
ARL & L3 TABLES
RULES TABLE
32b
VLAN TABLES
33b
20b
RMII
EPIC
ARL& L3 TABLES
21b
RULES TABLE
22b
VLAN TABLES
23b
16 Gbps CHANNEL
C
P
S
81
82
83
80
GBP
60
MMU/
CBM
71
70
ARL & L3 TABLES
21a
RULES TABLE
22a
VLAN TABLES
23a
EPIC
RMII
20a
CBP
50
EPIC
20c
RMII
ARL& L3 TABLES
21c
RULES TABLES
22c
VLAN TABLES
23c
IPIC
TABLES
91
NBP
92
90

Fig.3

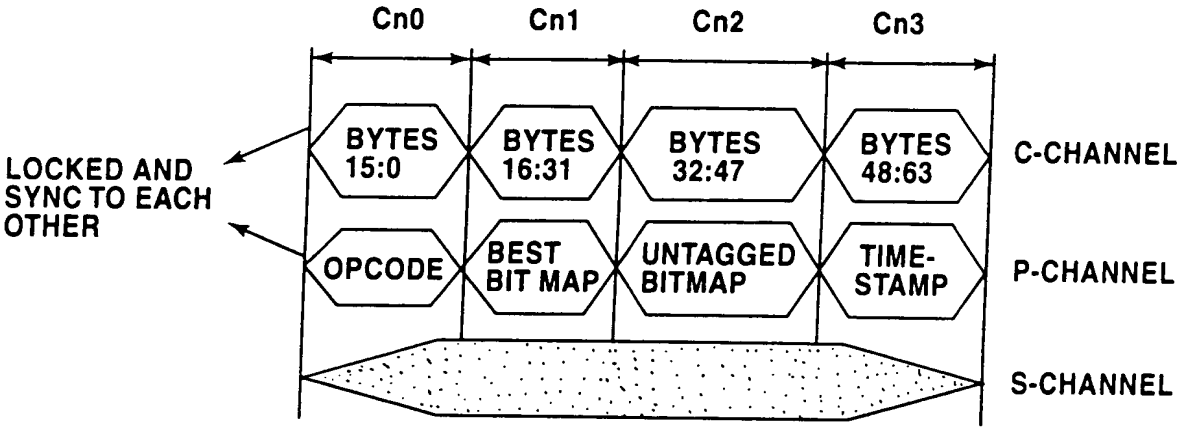

Cn0
Cn1
Cn2
Cn3
LOCKED AND SYNC TO EACH OTHER
BYTES 15:0
BYTES 16:31
BYTES 32:47
BYTES 48:63
C-CHANNEL
OPCODE
BEST BIT MAP
UNTAGGED BITMAP
TIME-STAMP
P-CHANNEL
S-CHANNEL

Fig.4a

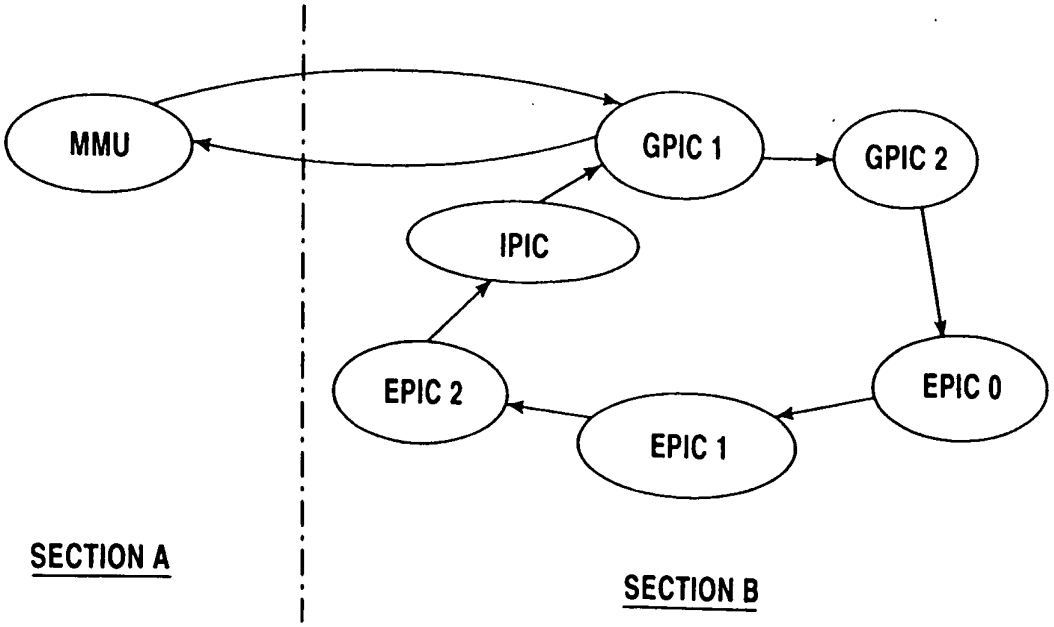

MMU
GPIC 1
GPIC 2
IPIC
EPIC 2
EPIC 0
EPIC 1
SECTION A
SECTION B

Fig.4b

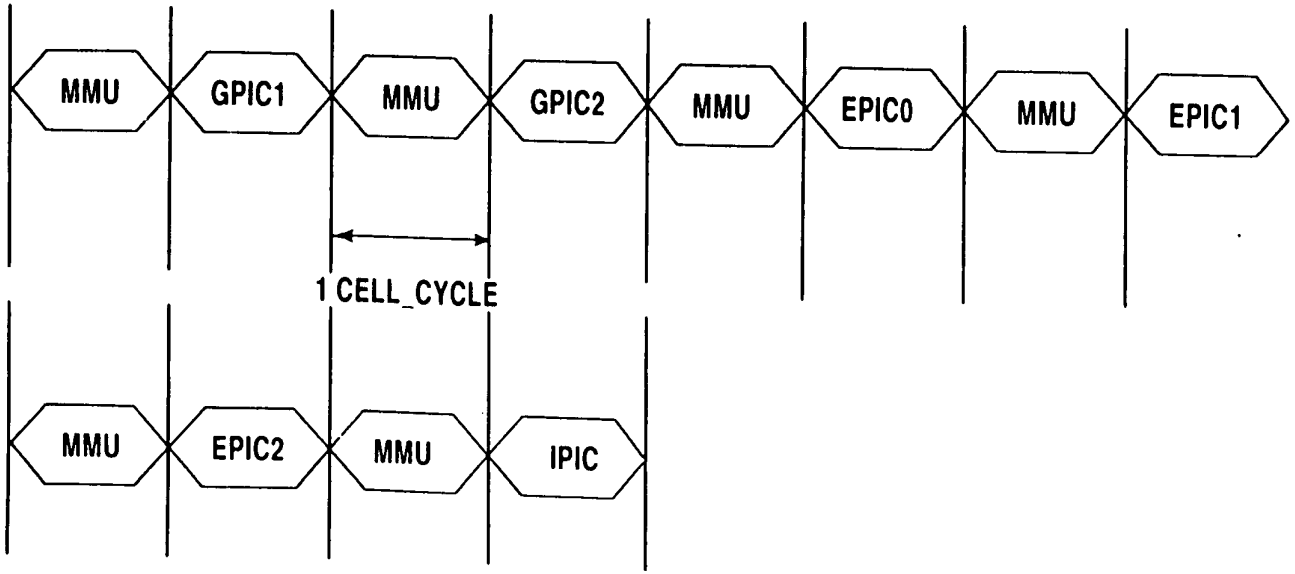

MMU
GPIC1
MMU
GPIC2
MMU
EPIC0
MMU
EPIC1
1 CELL_CYCLE
MMU
EPIC2
MMU
IPIC

Fig.5

PROTOCOL CHANNEL MESSAGES

30	28	26	24	22	20	18	16	14	12	10	8	6	4	2	0
OPC ODE	IP IPX	RESE RVED	NXT CELL	SRC DEST PORT			COS	J	S	E	CRC	P	O	LEN	

62	60	58	56	54	52	50	48	46	44	42	40	38	36	34	32
MODULE ID BITMAP															

30	28	26	24	22	20	18	16	14	12	10	8	6	4	2	0
R	Bc / Mc PORTBITMAP														

62	60	58	56	54	52	50	48	46	44	42	40	38	36	34	32
PF M	NEW IP CHECKSUM								M	MT-MOD ID		T	TGID	MOD OPCODE	c

30	28	26	24	22	20	18	16	14	12	10	8	6	4	2	0
U	UNTAGGED PORTBITMAP / SRC PORT NUMBER (bit0...5)														

62	60	58	56	54	52	50	48	46	44	42	40	38	36	34	32
RSVD		MATCHED FILTER		VLAN ID						SRC PORT			REMOTE PORT		

30	28	26	24	22	20	18	16	14	12	10	8	6	4	2	0
CPU OPCODES									TIME STAMP						

62	60	58	56	54	52	50	48	46	44	42	40	38	36	34	32
R	L3 PORT BITMAP														

Fig.6

SIDE BAND CHANNEL MESSAGES

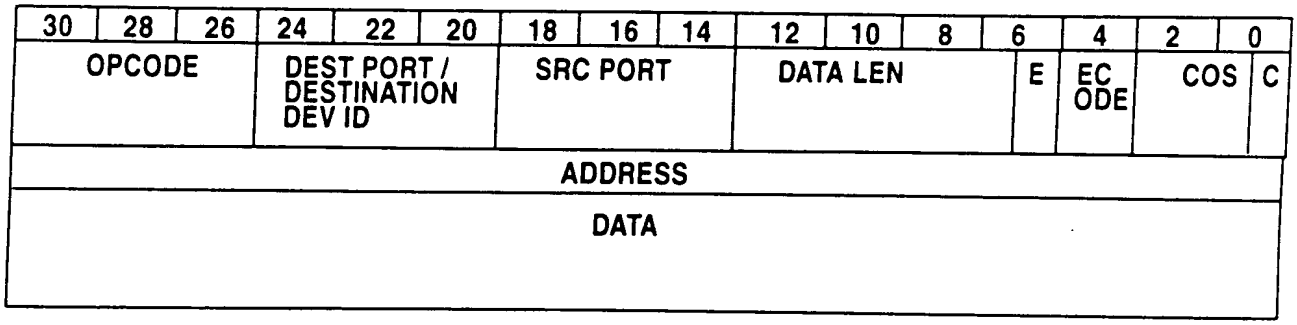


Fig.7

PRIOR ART

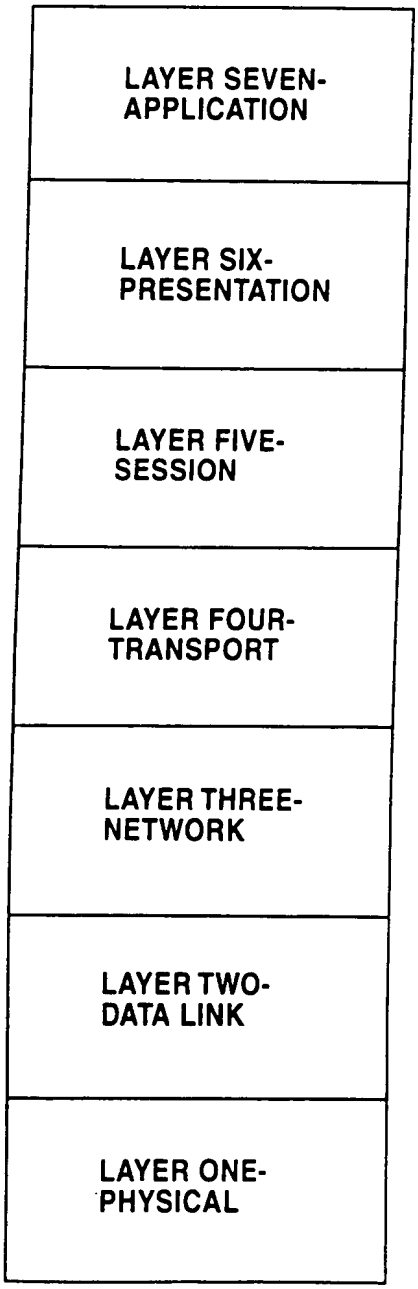
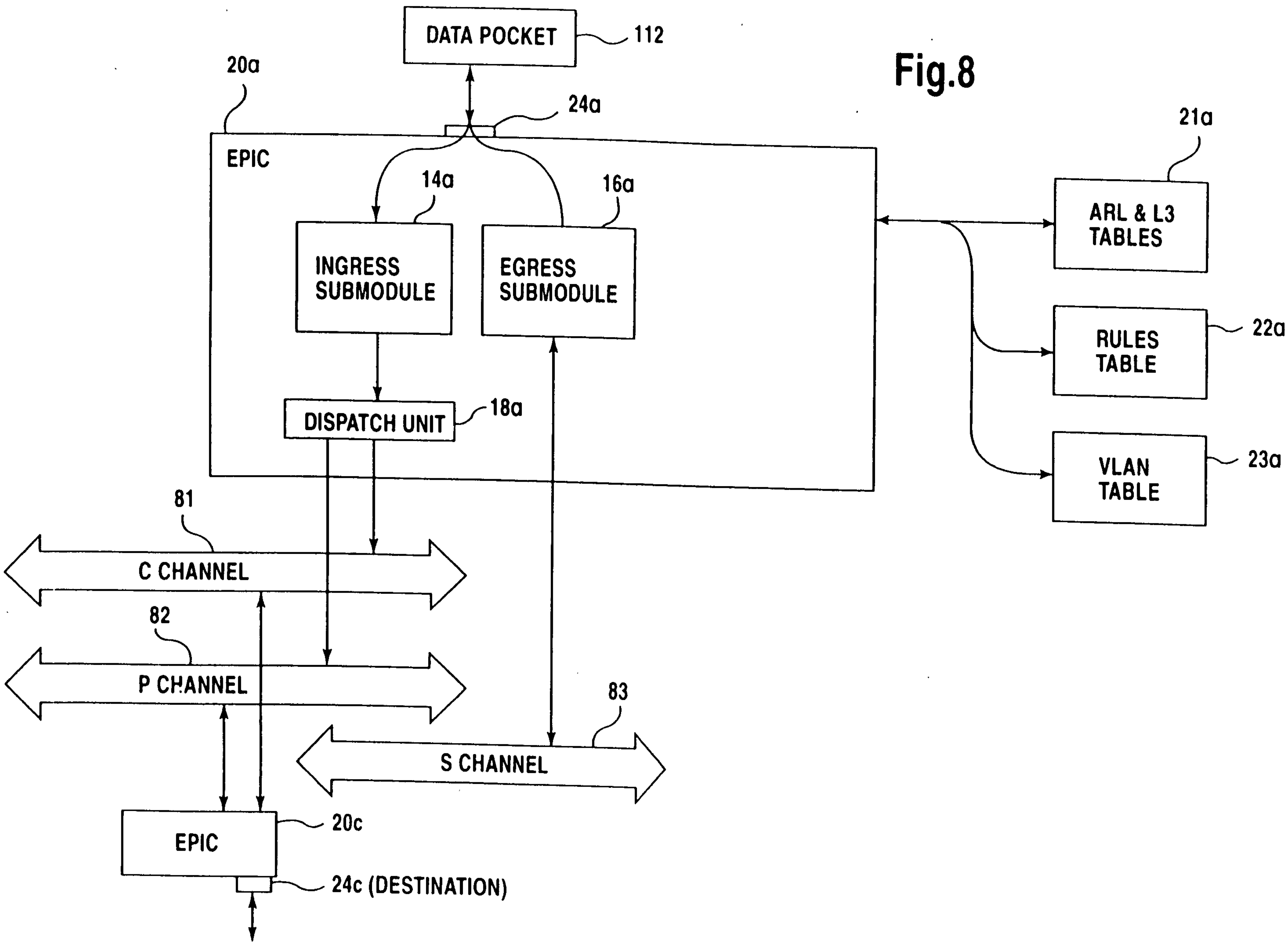


Fig.8



DATA POCKET
112
20a
24a
EPIC
14a
16a
INGRESS SUBMODULE
EGRESS SUBMODULE
DISPATCH UNIT
18a
21a
ARL & L3 TABLES
RULES TABLE
22a
VLAN TABLE
23a
81
C CHANNEL
82
P CHANNEL
83
S CHANNEL
EPIC
20c
24c (DESTINATION)

Fig.9

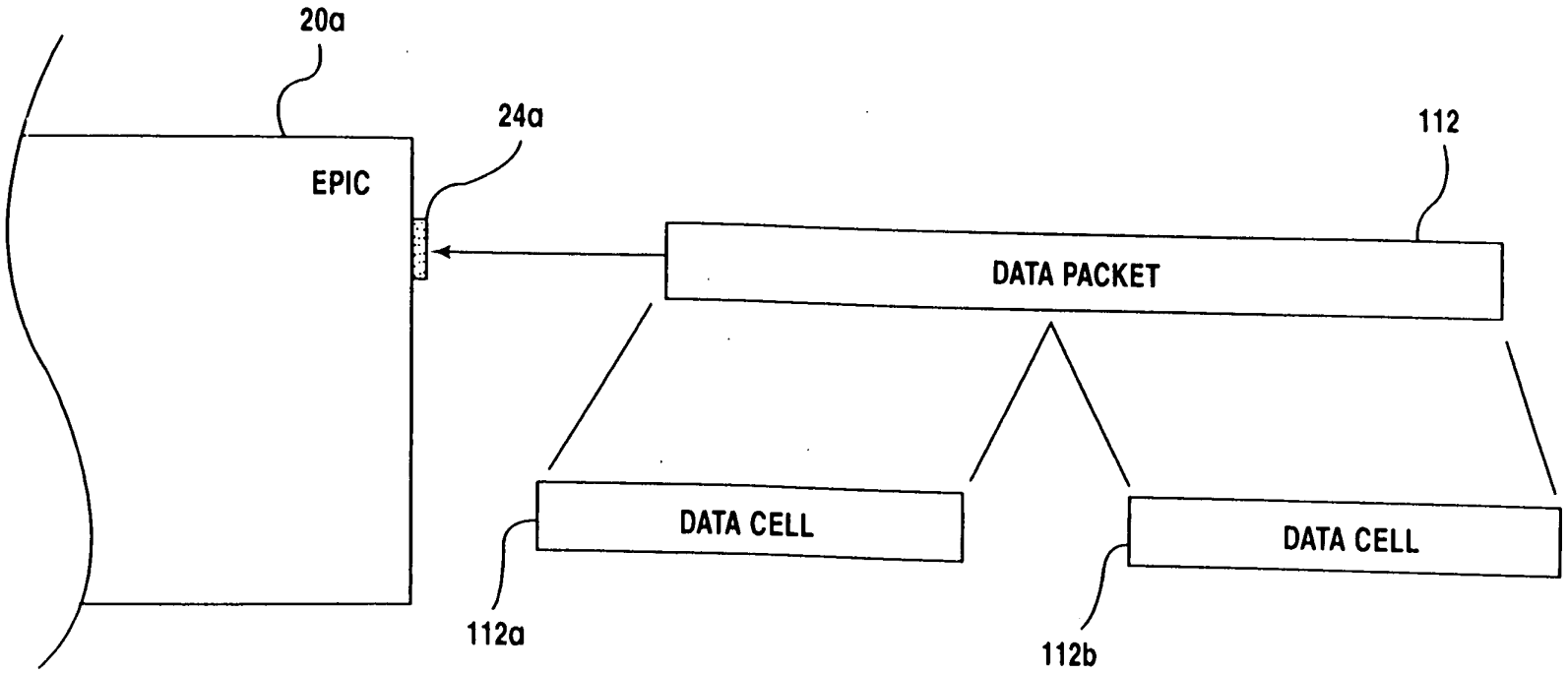


Fig.10

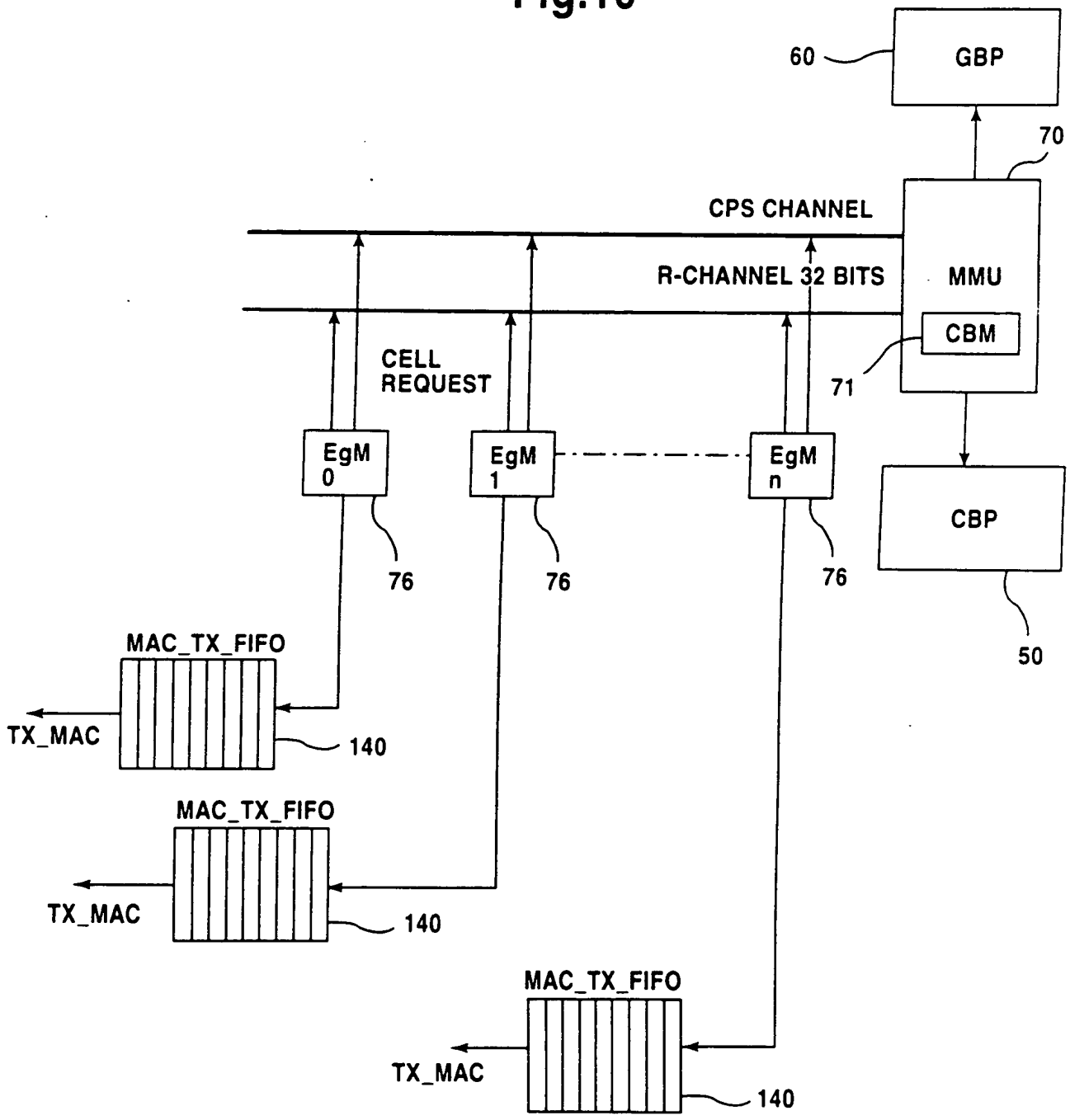

60
GBP
70
CPS CHANNEL
R-CHANNEL 32 BITS
MMU
CBM
71
CELL
REQUEST
EgM
0
EgM
1
EgM
n
CBP
50
76
76
76
MAC_TX_FIFO
TX_MAC
140
MAC_TX_FIFO
TX_MAC
140
MAC_TX_FIFO
TX_MAC
140

Fig.11

LINE 0 ⟶	FC I LC I BC/MC I Cpy_cnt (5b) I Cell_length (7b) I CRC (2b) I NC_header (16b) I Src Count (6) I IPX I IP I I Time_Stamp (14b) I O bits (2b) I P I NextCellLen (2b) I CpuOpcode (4b) I Cell_data (0-9B)
LINE 1 ⟶	Cell_data (10-27) Bytes
LINE 2 ⟶	Cell_data (28-45) Bytes
LINE 3 ⟶	Cell_data (45-63) Bytes

Fig.12

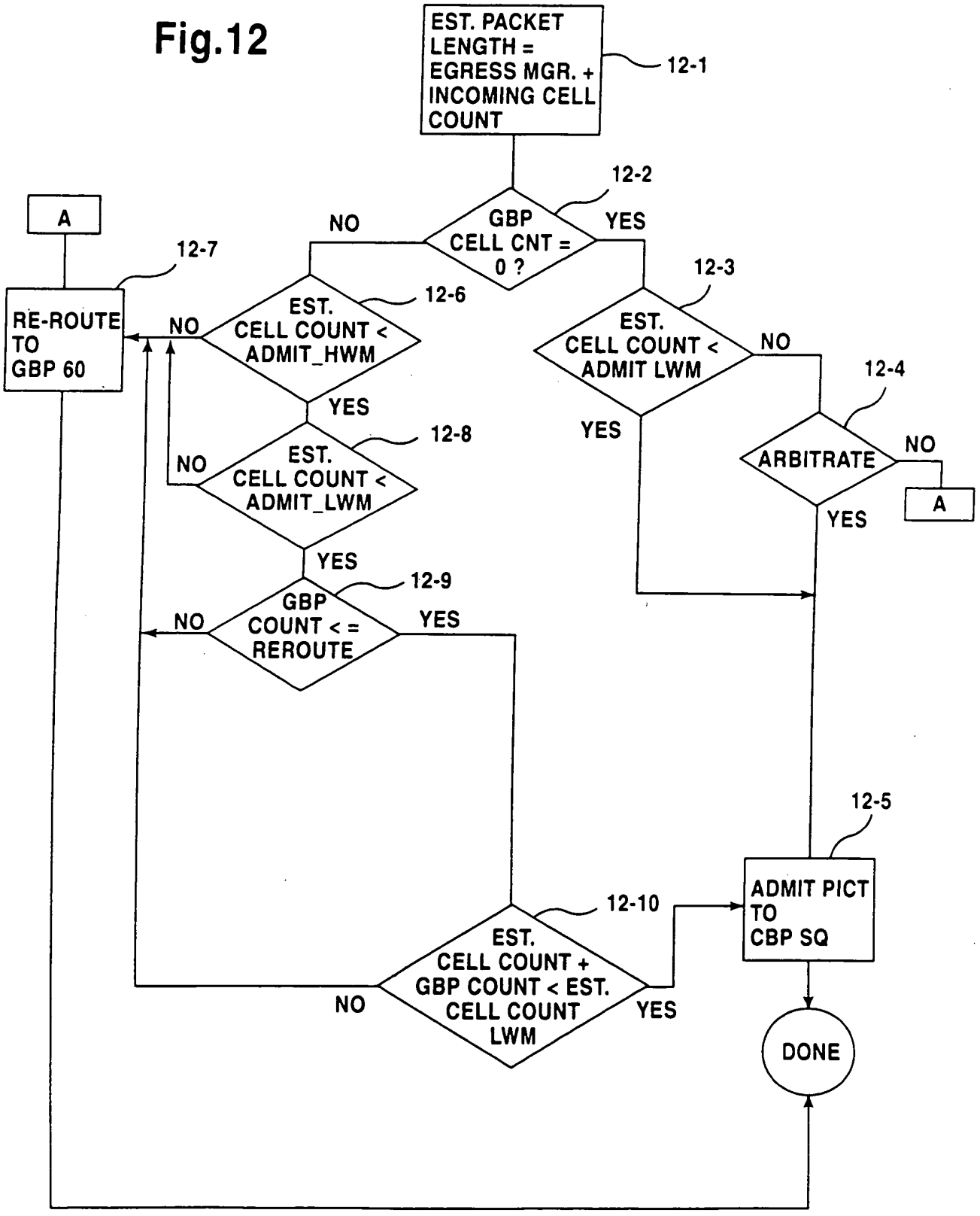

EST. PACKET LENGTH = EGRESS MGR. + INCOMING CELL COUNT
12-1
GBP CELL CNT = 0 ?
12-2
NO
YES
A
12-7
RE-ROUTE TO GBP 60
EST. CELL COUNT < ADMIT_HWM
12-6
NO
YES
EST. CELL COUNT < ADMIT_LWM
12-3
NO
YES
ARBITRATE
12-4
NO
A
YES
EST. CELL COUNT < ADMIT_LWM
12-8
NO
YES
GBP COUNT < = REROUTE
12-9
NO
YES
ADMIT PICT TO CBP SQ
12-5
EST. CELL COUNT + GBP COUNT < EST. CELL COUNT LWM
12-10
NO
YES
DONE

Fig.13

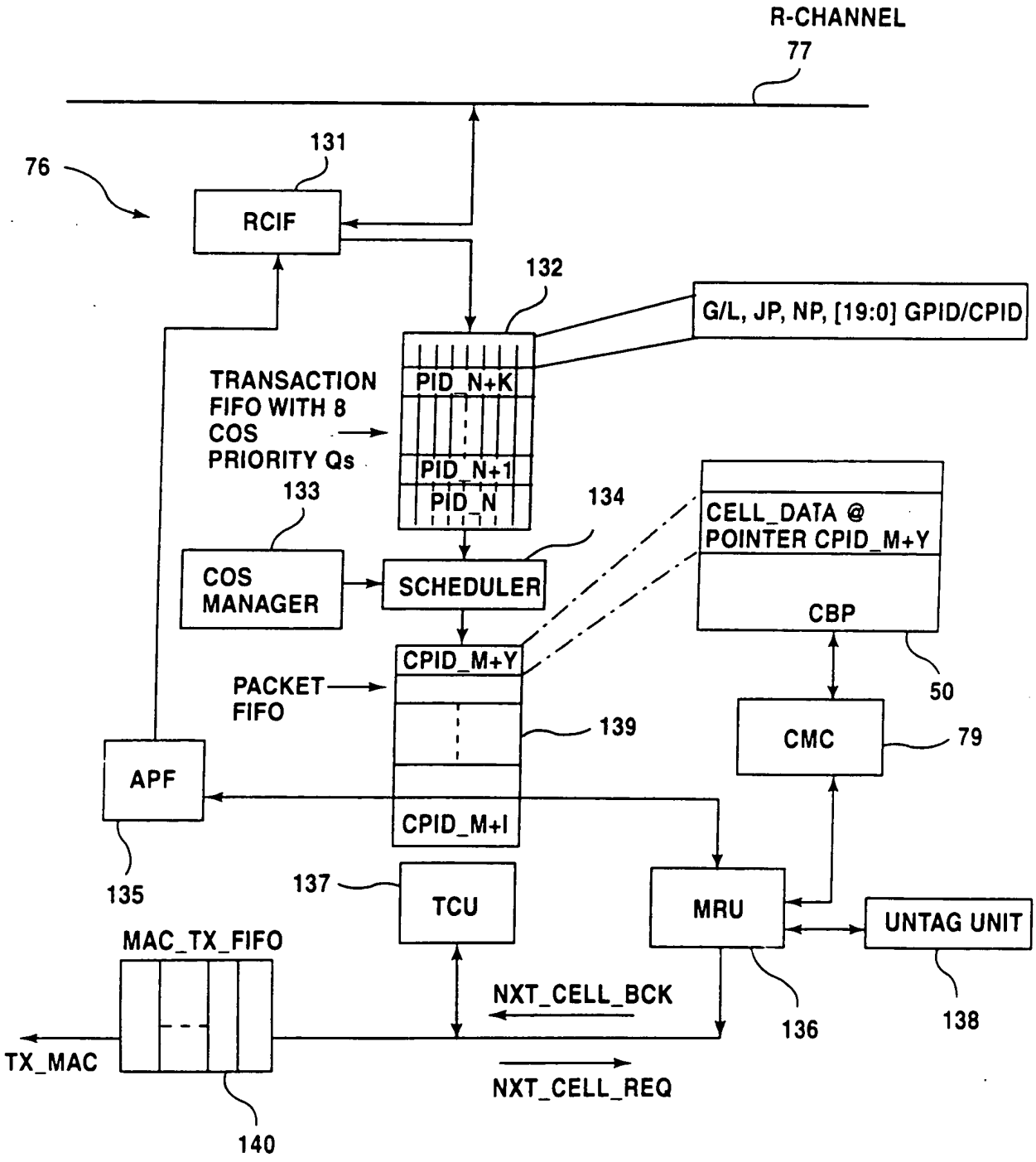

R-CHANNEL
77
76
131
RCIF
132
G/L, JP, NP, [19:0] GPID/CPID
TRANSACTION FIFO WITH 8 COS PRIORITY Qs
PID_N+K
PID_N+1
PID_N
133
134
CELL_DATA @ POINTER CPID_M+Y
CBP
50
COS MANAGER
SCHEDULER
CPID_M+Y
PACKET FIFO
139
CMC
79
APF
CPID_M+I
135
137
TCU
MRU
UNTAG UNIT
MAC_TX_FIFO
NXT_CELL_BCK
136
138
TX_MAC
NXT_CELL_REQ
140

Fig.14

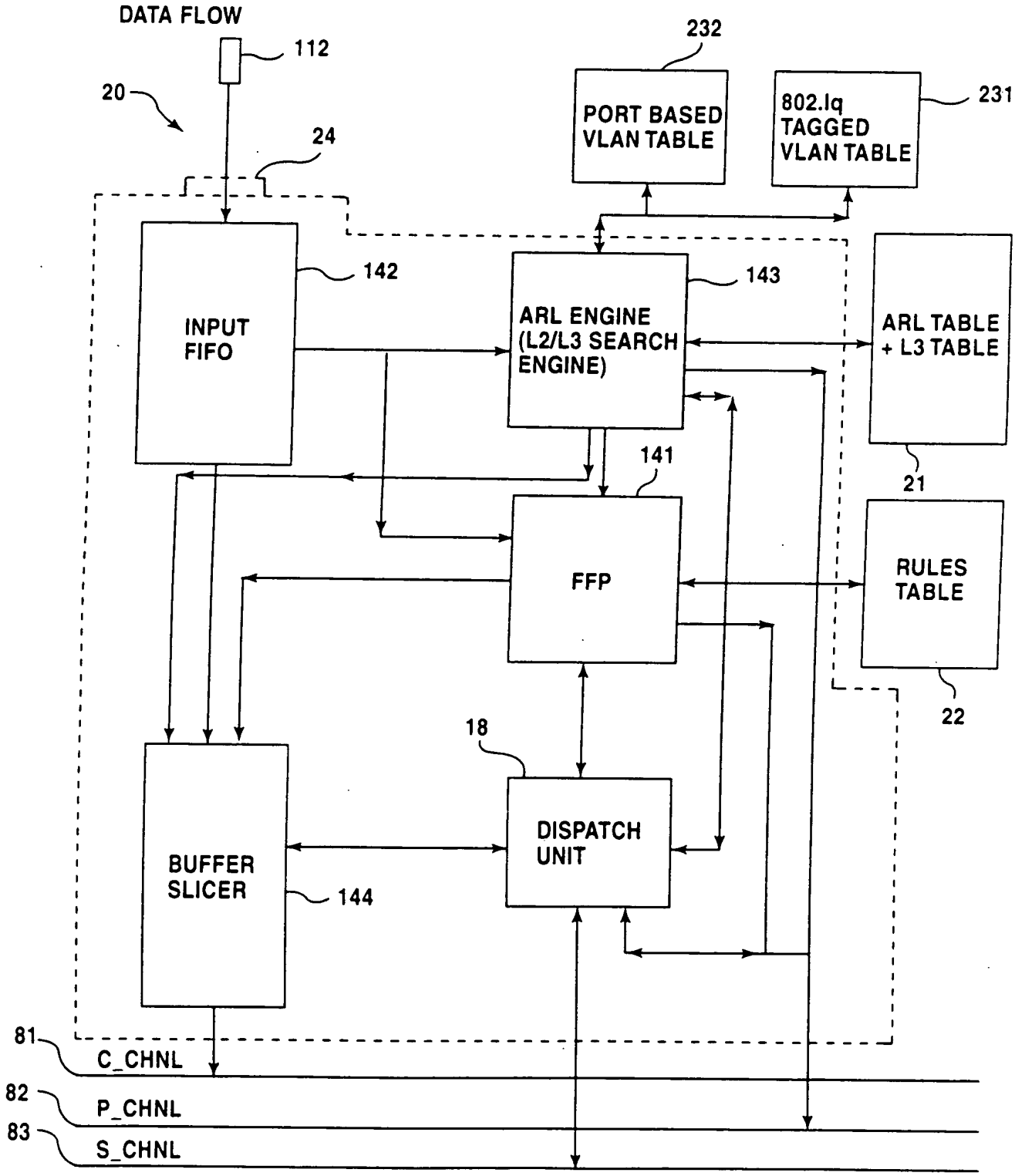

DATA FLOW
112
20
24
232
PORT BASED VLAN TABLE
802.1q TAGGED VLAN TABLE
231
INPUT FIFO
142
ARL ENGINE (L2/L3 SEARCH ENGINE)
143
ARL TABLE + L3 TABLE
21
141
FFP
RULES TABLE
22
18
DISPATCH UNIT
BUFFER SLICER
144
81
C_CHNL
82
P_CHNL
83
S_CHNL

Fig.15

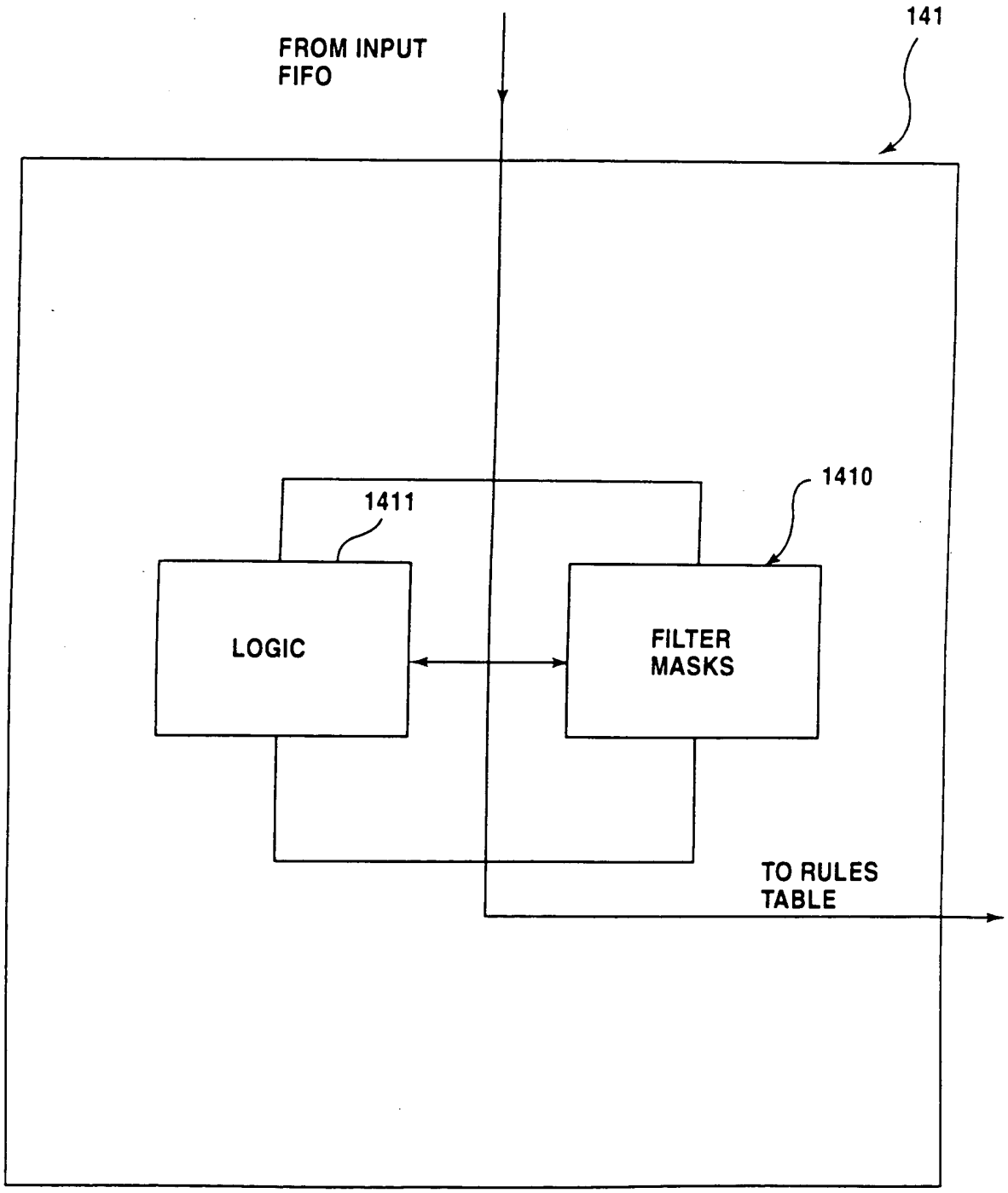

141
FROM INPUT FIFO
1410
1411
LOGIC
FILTER MASKS
TO RULES TABLE

Fig.16

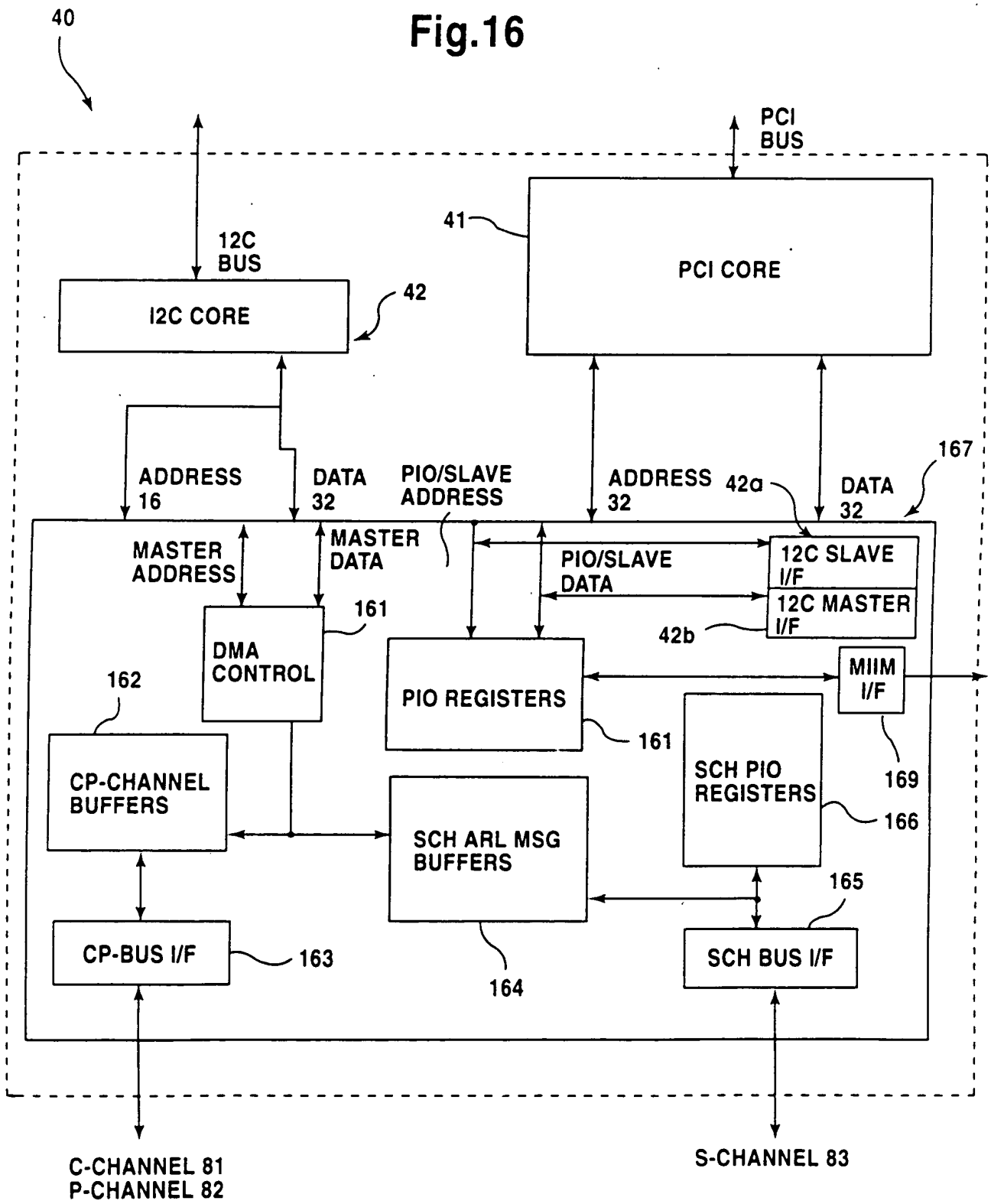

40
PCI BUS
41
PCI CORE
12C BUS
I2C CORE
42
ADDRESS 16
DATA 32
PIO/SLAVE ADDRESS
ADDRESS 32
42a
DATA 32
167
MASTER ADDRESS
MASTER DATA
PIO/SLAVE DATA
12C SLAVE I/F
12C MASTER I/F
42b
161
DMA CONTROL
PIO REGISTERS
MIIM I/F
162
CP-CHANNEL BUFFERS
SCH PIO REGISTERS
169
166
SCH ARL MSG BUFFERS
165
CP-BUS I/F
163
164
SCH BUS I/F
C-CHANNEL 81
P-CHANNEL 82
S-CHANNEL 83

Fig.17 FFP PROGRAMMING FLOW CHART

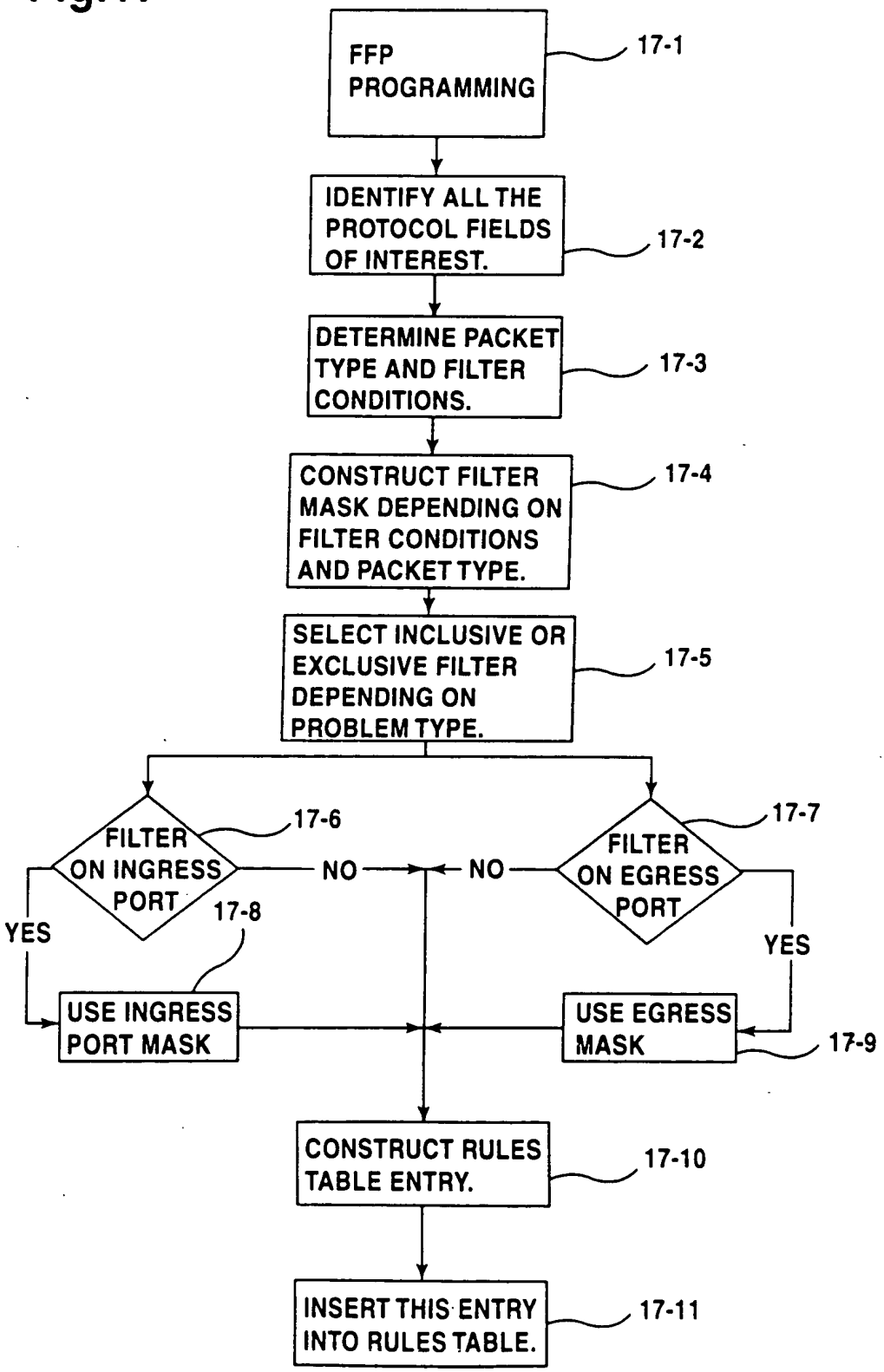


Fig.18

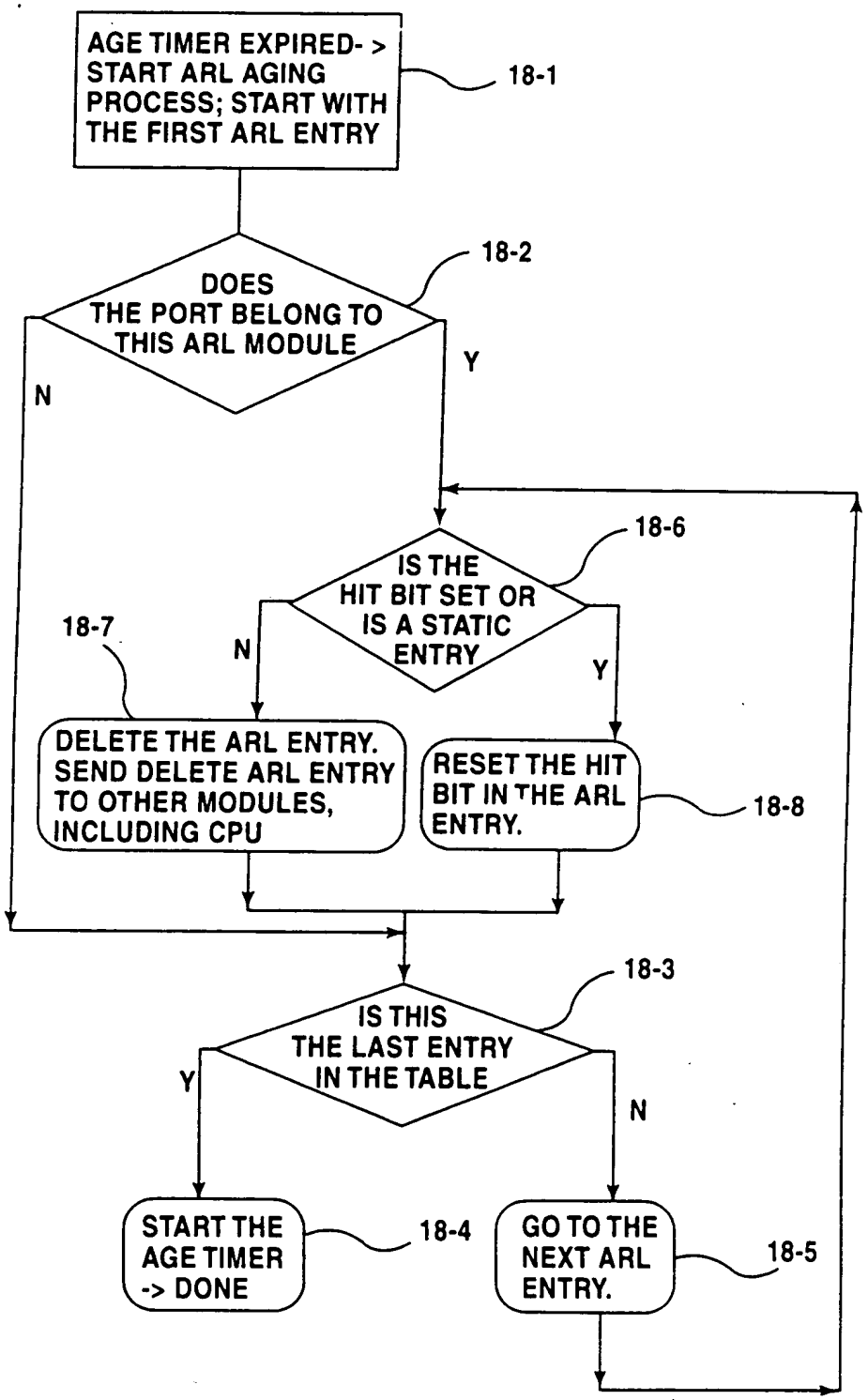
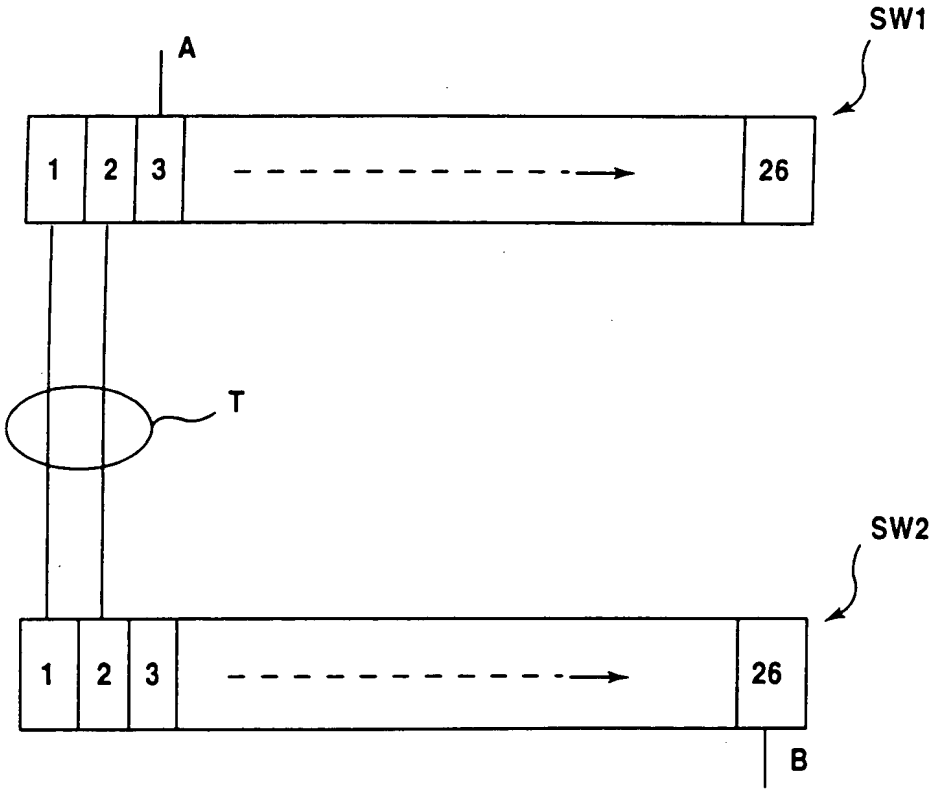

AGE TIMER EXPIRED- > START ARL AGING PROCESS; START WITH THE FIRST ARL ENTRY
18-1
DOES THE PORT BELONG TO THIS ARL MODULE
18-2
Y
N
IS THE HIT BIT SET OR IS A STATIC ENTRY
18-6
N
Y
18-7
DELETE THE ARL ENTRY. SEND DELETE ARL ENTRY TO OTHER MODULES, INCLUDING CPU
RESET THE HIT BIT IN THE ARL ENTRY.
18-8
IS THIS THE LAST ENTRY IN THE TABLE
18-3
Y
N
START THE AGE TIMER -> DONE
18-4
GO TO THE NEXT ARL ENTRY.
18-5

Fig.19



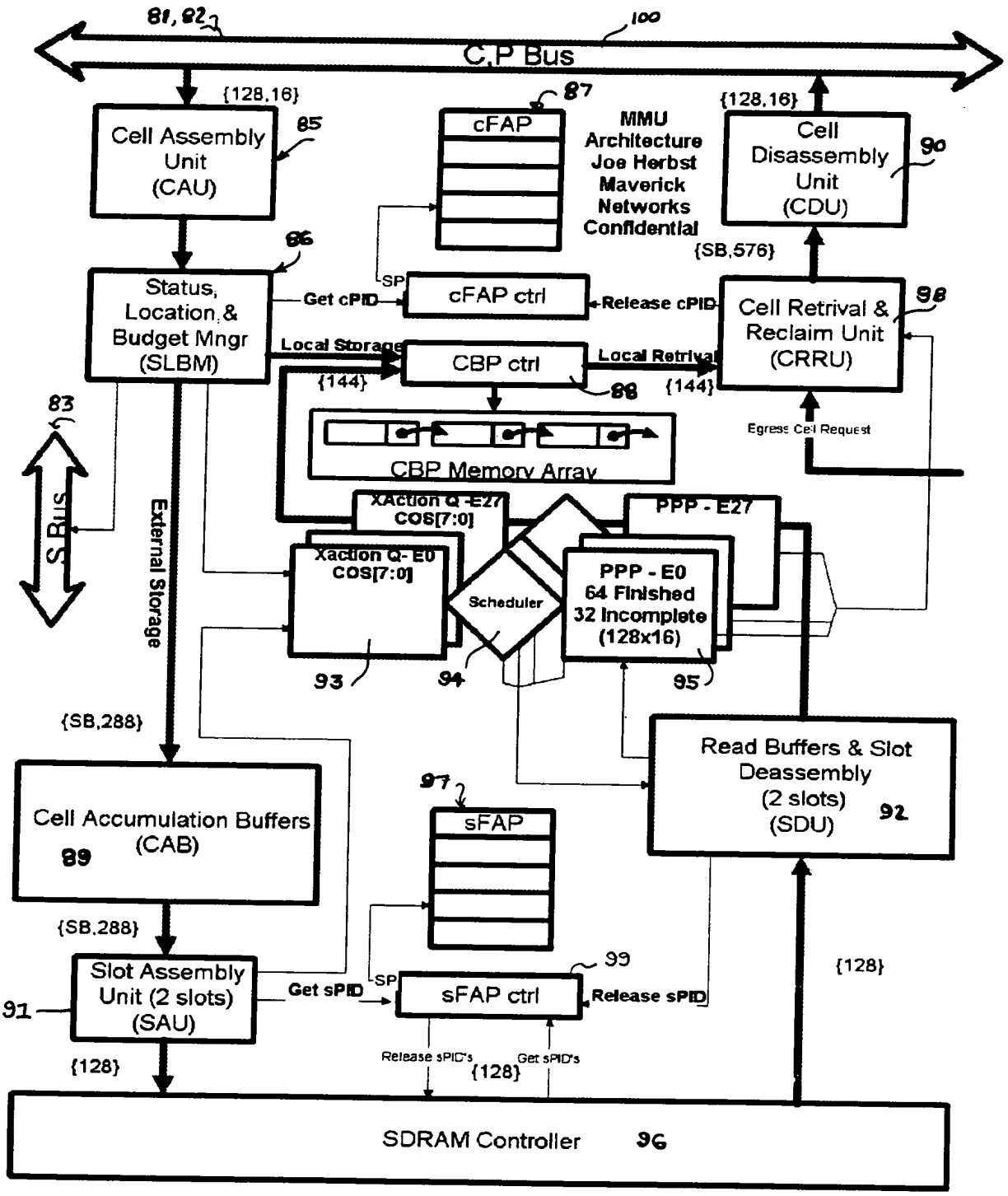

81, 82
100
C,P Bus
{128,16}
87
cFAP
MMU
Architecture
Joe Herbst
Maverick
Networks
Confidential
{128,16}
Cell Assembly Unit (CAU)
85
Cell Disassembly Unit (CDU)
90
{SB,576}
86
Status, Location & Budget Mngr (SLBM)
SP
Get cPID
cFAP ctrl
Release cPID
Cell Retrival & Reclaim Unit (CRRU)
98
Local Storage
CBP ctrl
Local Retrival
{144}
88
{144}
83
CBP Memory Array
Egress Cell Request
SBus
External Storage
XAction Q -E27 COS[7:0]
PPP - E27
Xaction Q- E0 COS[7:0]
Scheduler
PPP - E0
64 Finished
32 Incomplete
(128x16)
93
94
95
{SB,288}
Read Buffers & Slot Deassembly (2 slots) (SDU)
92
97
sFAP
Cell Accumulation Buffers (CAB)
89
{SB,288}
99
Slot Assembly Unit (2 slots) (SAU)
91
Get sPID
SP
sFAP ctrl
Release sPID
{128}
Release sPID's
Get sPID's
{128}
{128}
SDRAM Controller
96

FIGURE 20

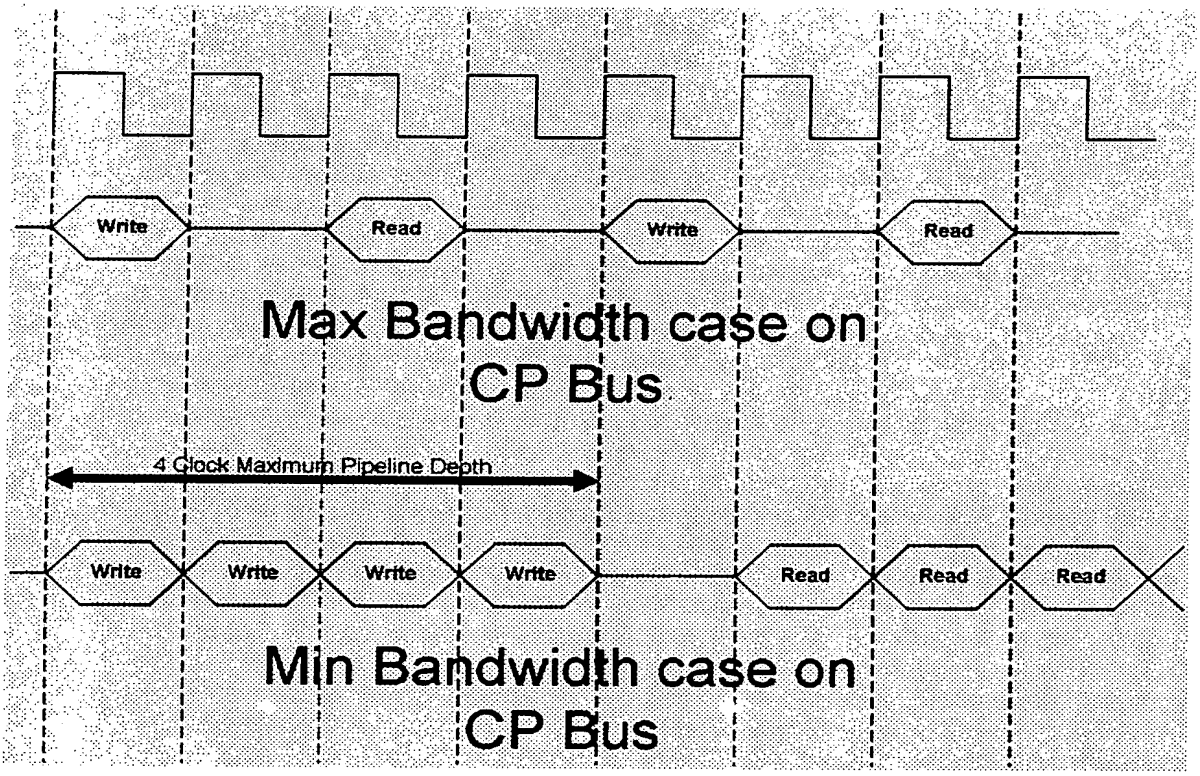

Write
Read
Write
Read
Max Bandwidth case on CP Bus
4 Clock Maximum Pipeline Depth
Write
Write
Write
Write
Read
Read
Read
Min Bandwidth case on CP Bus

Figure 21

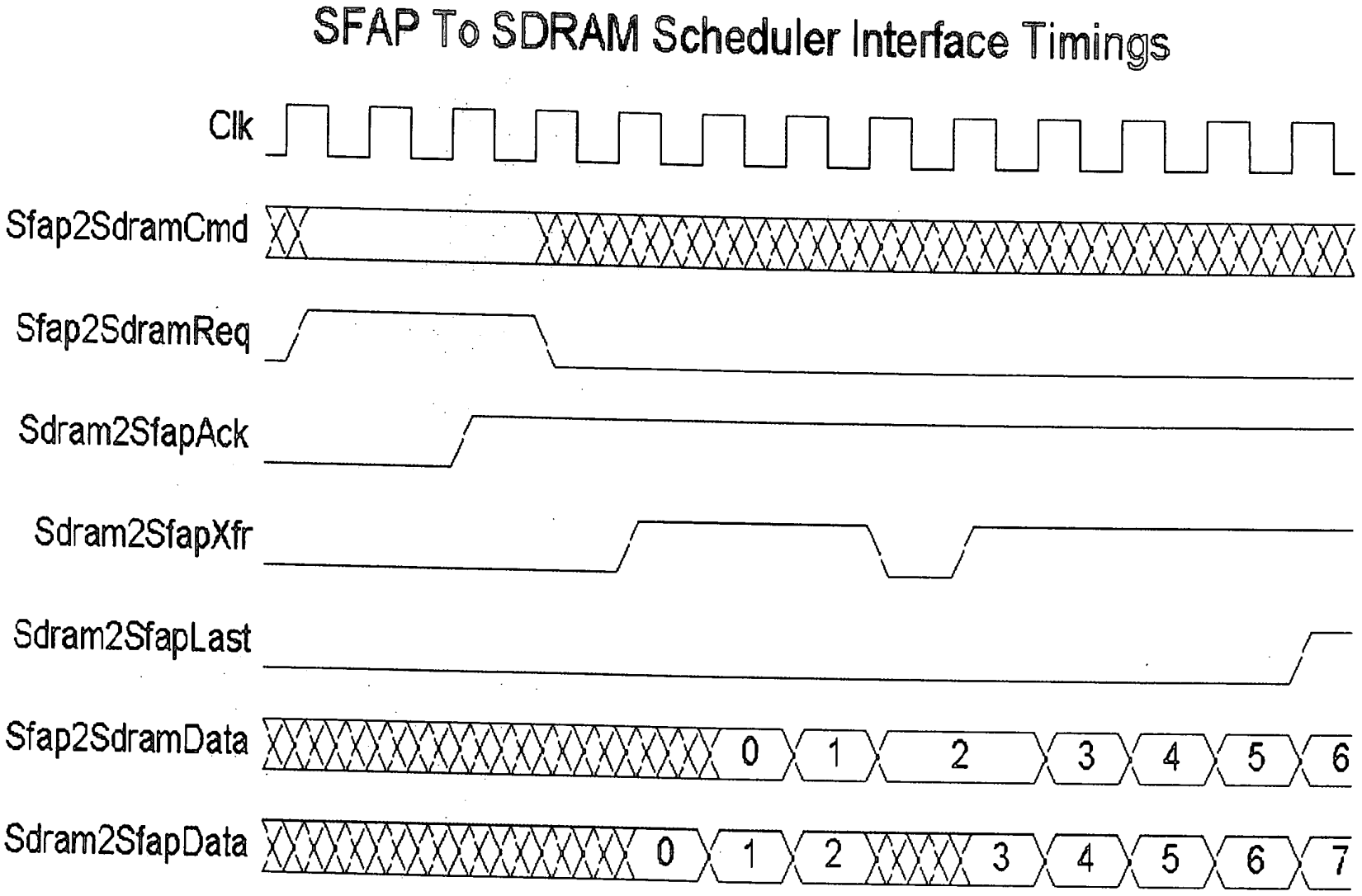
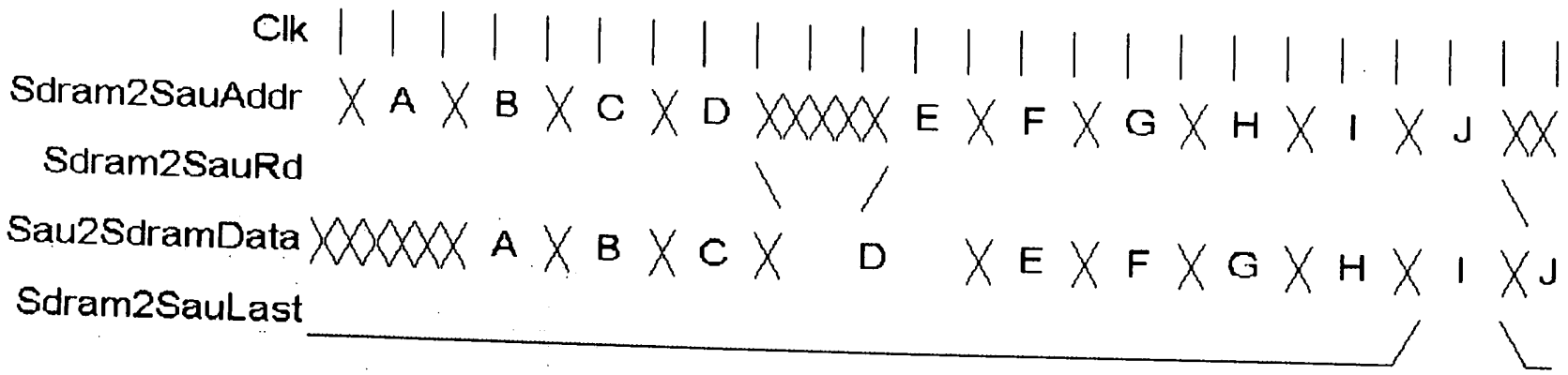

SFAP To SDRAM Scheduler Interface Timings
Clk
Sfap2SdramCmd
Sfap2SdramReq
Sdram2SfapAck
Sdram2SfapXfr
Sdram2SfapLast
Sfap2SdramData
0
1
2
3
4
5
6
Sdram2SfapData
0
1
2
3
4
5
6
7

FIGURE 22

FIGURE 23

SAU to SDRAM Scheduler Data Transfer



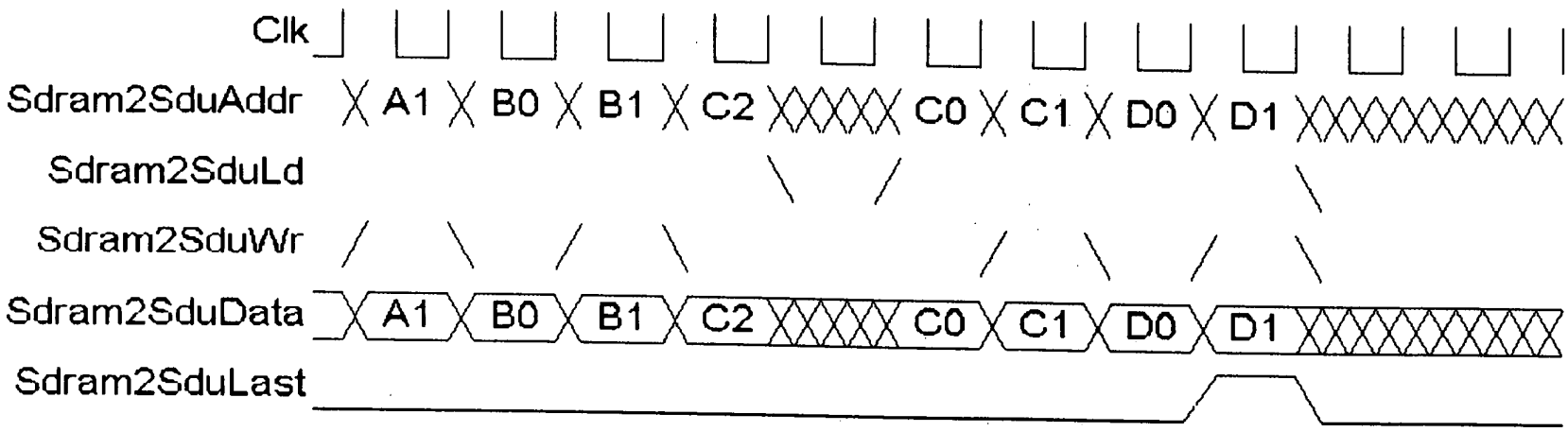
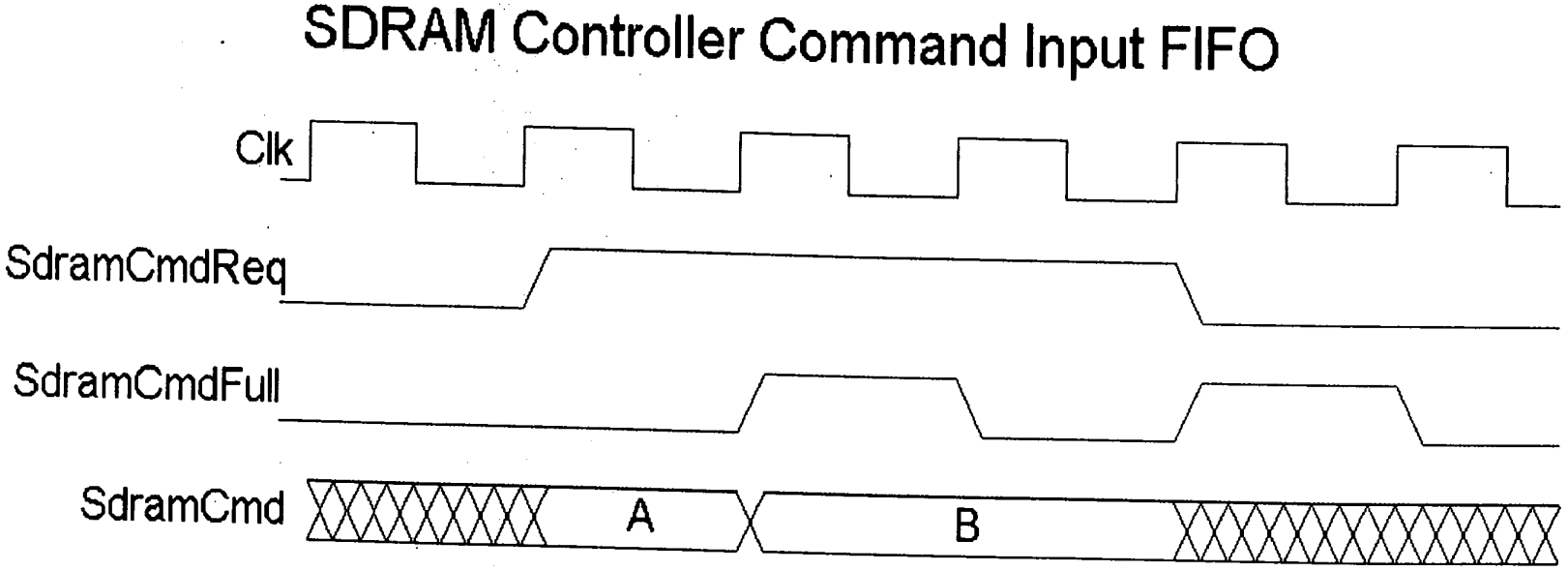

SDRAM Scheduler to SDU Data Transfer
Clk
Sdram2SduAddr
A1
B0
B1
C2
C0
C1
D0
D1
Sdram2SduLd
Sdram2SduWr
Sdram2SduData
A1
B0
B1
C2
C0
C1
D0
D1
Sdram2SduLast

FIGURE 24

FIGURE 25

SDRAM Controller Interface Timing



SDRAM Controller Data Write FIFO

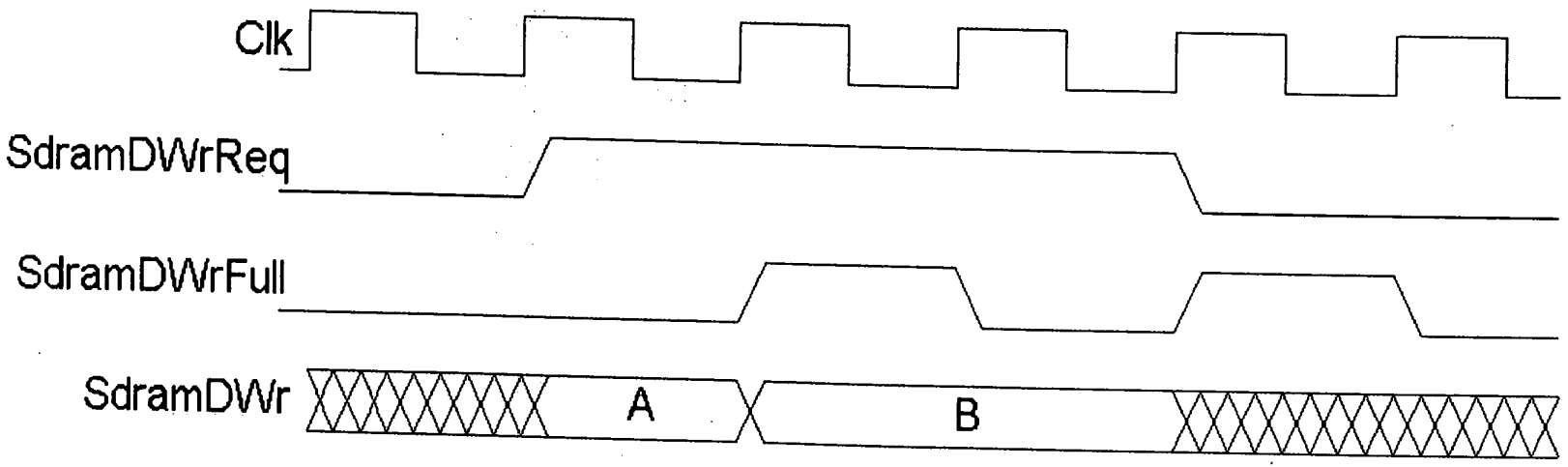
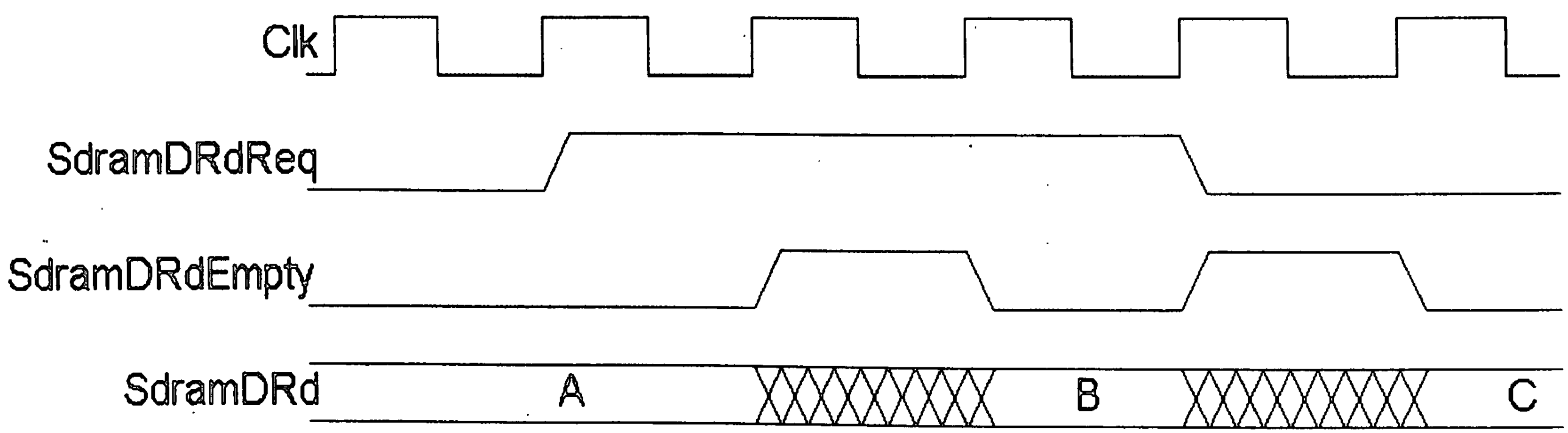


FIGURE 26

FIGURE 27

SDRAM Controller Data Read FIFO



Field	Left	Right	Bits
Src	310	306	5
CPUOpcode	305	302	4
BC/MC Bitmap	301	270	32
Cos	269	267	3
P	266	266	1
FC (S)	265	265	1
LC (E)	264	264	1
CRC	263	262	2
Len (0 = 64)	261	256	6
O	255	254	2
BC/MC	253	253	1
Copy Count (0 = 32)	252	248	5
Untagged Bitmap	247	216	32
IP	215	215	1
IPX	214	214	1
Time Stamp	213	200	14
Cell Data Bytes 24-0	199	0	200
Total			311

FIGURE 28

Cell Size	SAU Words	SDRAM Words
0 0	1	2
0 1	1	3
1 0	2	4
1 1	2	5

Figure 29

Field	Left	Right	Bits	First Only
Last Slot	313	313	1	X
Next Slot ID	312	297	16	X
Copy Count	296	292	5	X
CPUOpcode	291	288	4	
Cell Size	287	286	2	
P	285	285	1	
FC	284	284	1	
LC	283	283	1	
CRC	282	281	2	
Len	280	275	6	
O	274	273	2	
BC/MC	272	272	1	
IP	271	271	1	
IPX	270	270	1	
Time Stamp	269	256	14	

Figure 30

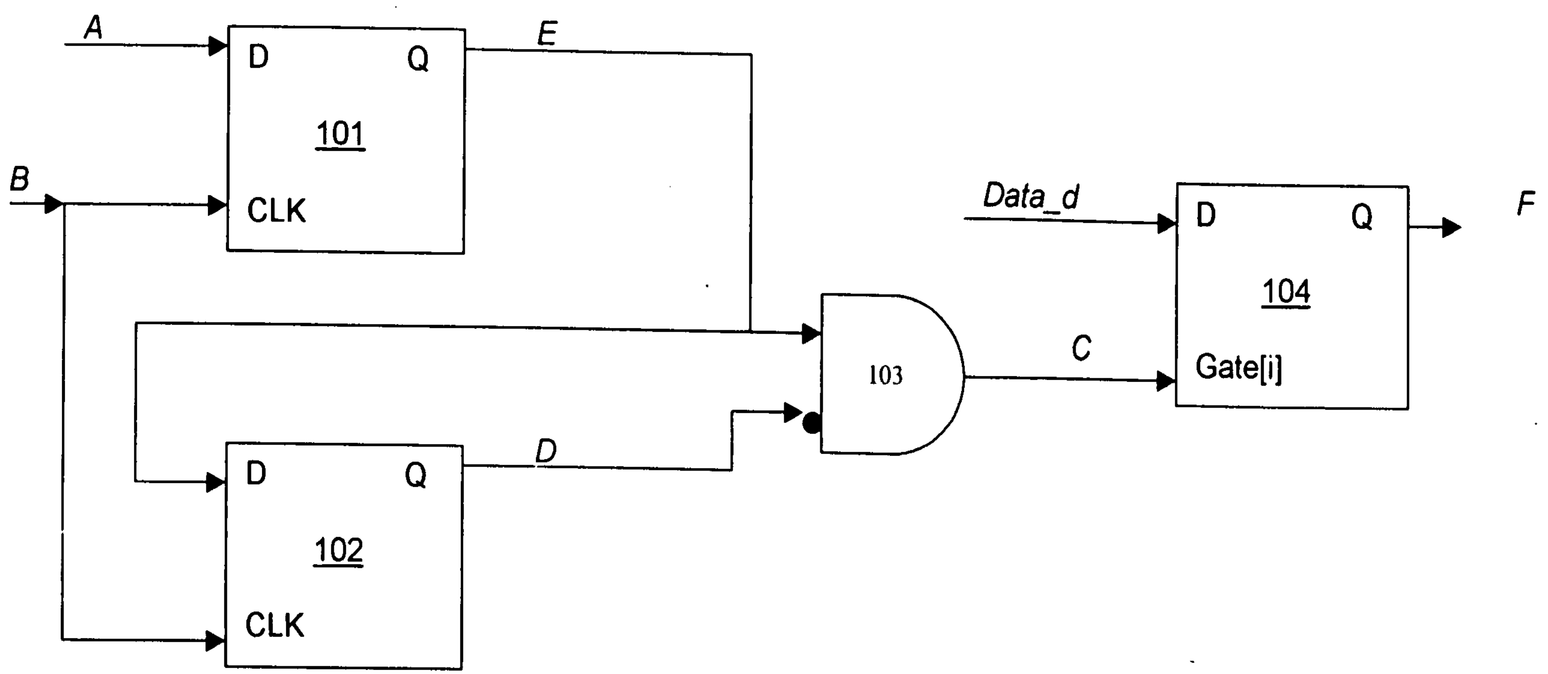
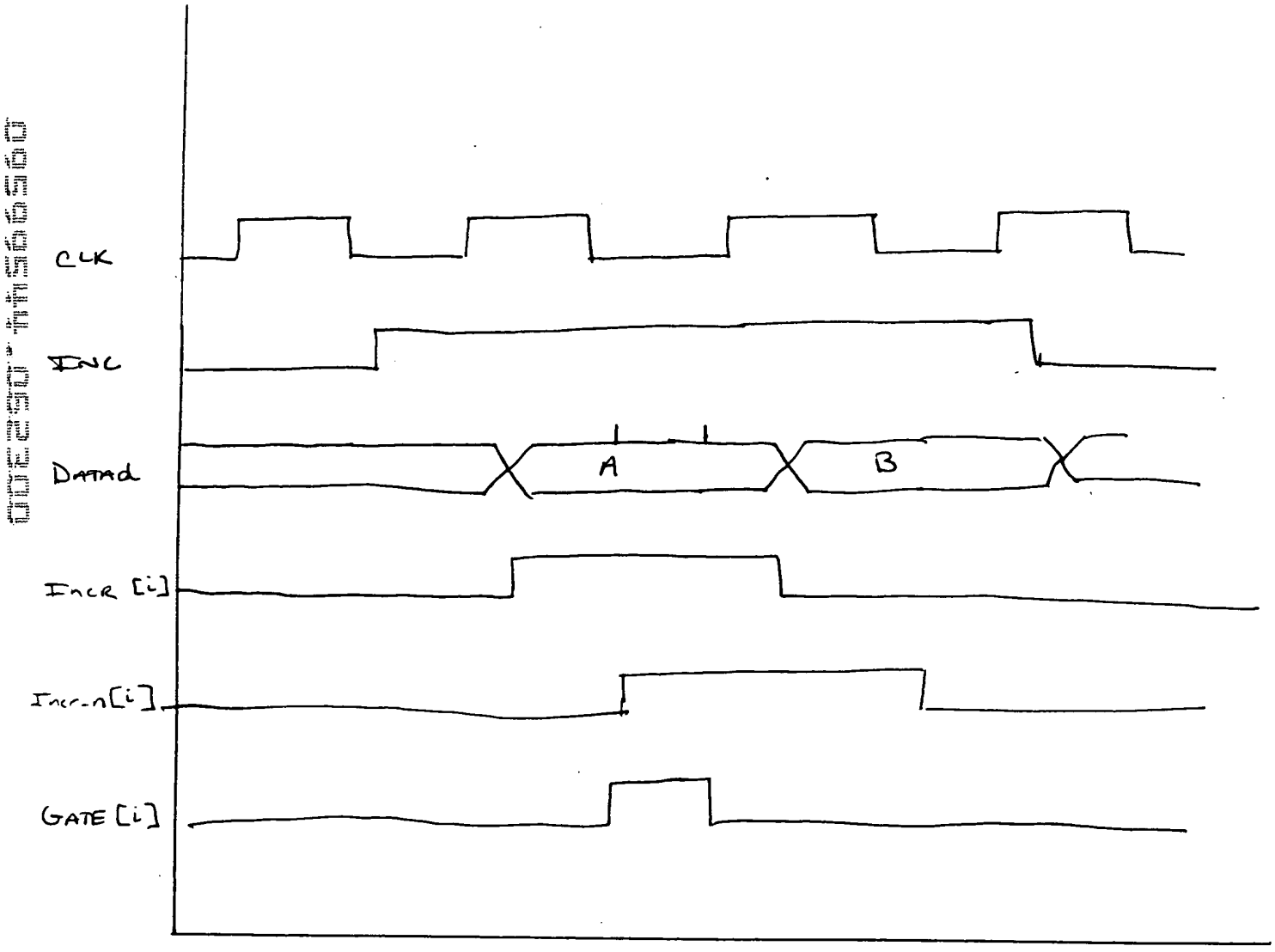


Figure 31

Figure 32



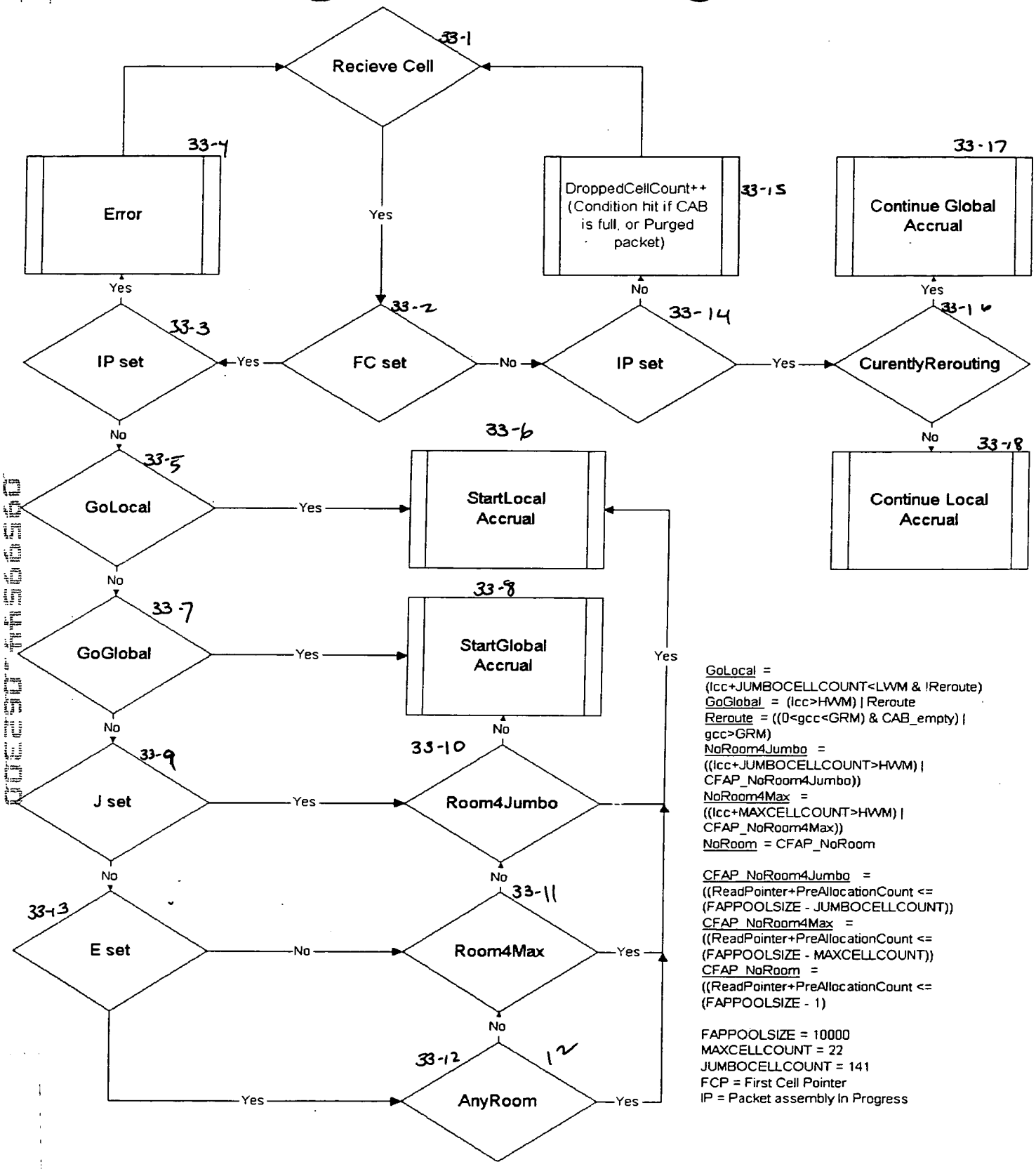

Recieve Cell
33-1
Error
33-4
DroppedCellCount++ (Condition hit if CAB is full, or Purged packet)
33-15
Continue Global Accrual
33-17
Yes
No
IP set
33-3
FC set
33-2
IP set
33-14
CurentlyRerouting
33-16
GoLocal
33-5
StartLocal Accrual
33-6
Continue Local Accrual
33-18
GoGlobal
33-7
StartGlobal Accrual
33-8
J set
33-9
Room4Jumbo
33-10
E set
33-13
Room4Max
33-11
AnyRoom
33-12
GoLocal = (Icc+JUMBOCELLCOUNT<LWM & !Reroute)
GoGlobal = (Icc>HWM) | Reroute
Reroute = ((0<gcc<GRM) & CAB_empty) | gcc>GRM)
NoRoom4Jumbo = ((Icc+JUMBOCELLCOUNT>HWM) | CFAP_NoRoom4Jumbo))
NoRoom4Max = ((Icc+MAXCELLCOUNT>HWM) | CFAP_NoRoom4Max))
NoRoom = CFAP_NoRoom
CFAP_NoRoom4Jumbo = ((ReadPointer+PreAllocationCount <= (FAPPOOLSIZE - JUMBOCELLCOUNT))
CFAP_NoRoom4Max = ((ReadPointer+PreAllocationCount <= (FAPPOOLSIZE - MAXCELLCOUNT))
CFAP_NoRoom = ((ReadPointer+PreAllocationCount <= (FAPPOOLSIZE - 1)
FAPPOOLSIZE = 10000
MAXCELLCOUNT = 22
JUMBOCELLCOUNT = 141
FCP = First Cell Pointer
IP = Packet assembly In Progress

FIGURE 33

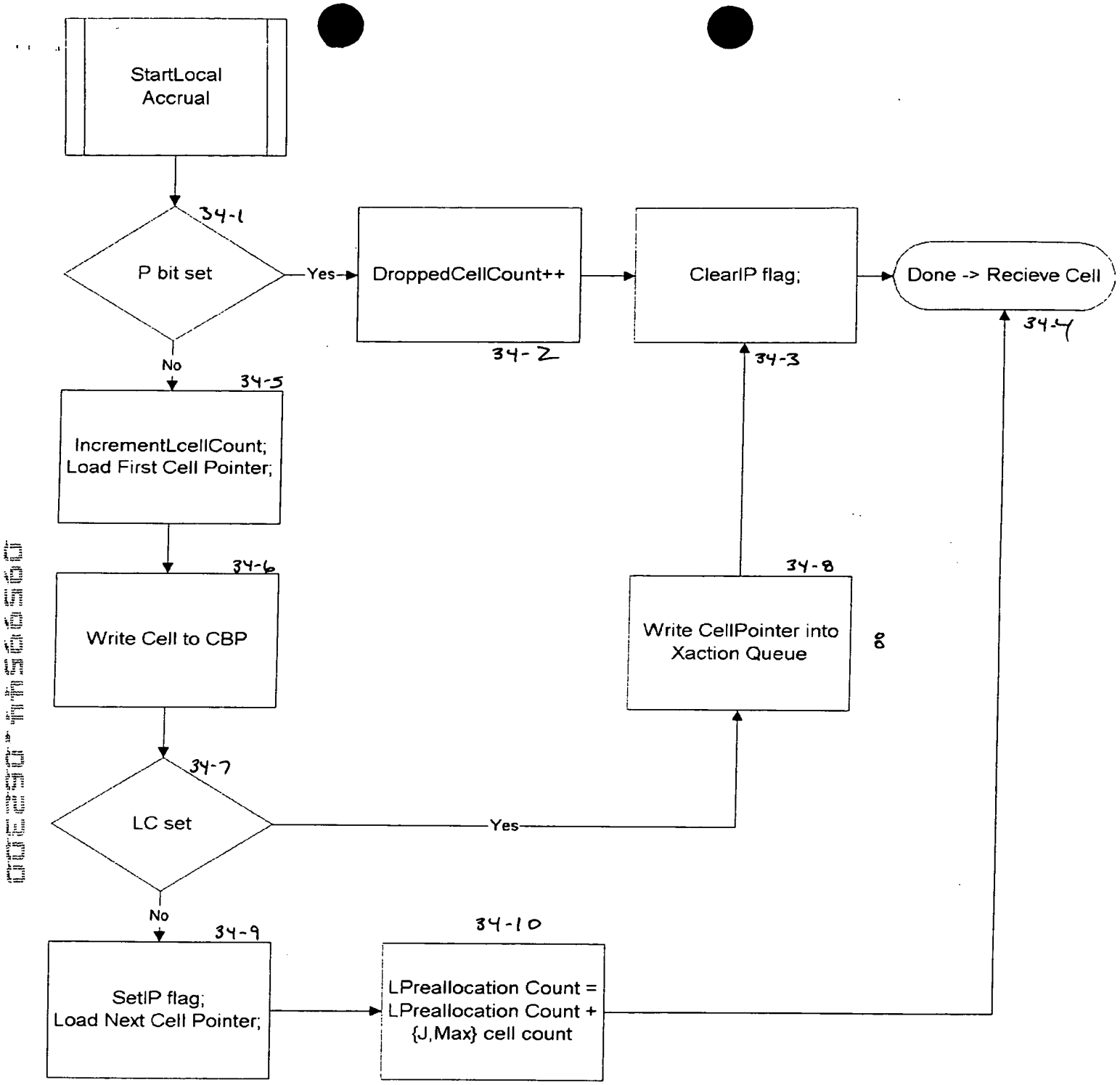

StartLocal Accrual
34-1
P bit set
Yes
DroppedCellCount++
34-2
ClearIP flag;
34-3
Done -> Recieve Cell
34-4
No
34-5
IncrementLcellCount;
Load First Cell Pointer;
34-6
Write Cell to CBP
34-7
LC set
Yes
34-8
Write CellPointer into Xaction Queue
No
34-9
SetIP flag;
Load Next Cell Pointer;
34-10
LPreallocation Count = LPreallocation Count + {J,Max} cell count

FIGURE 34

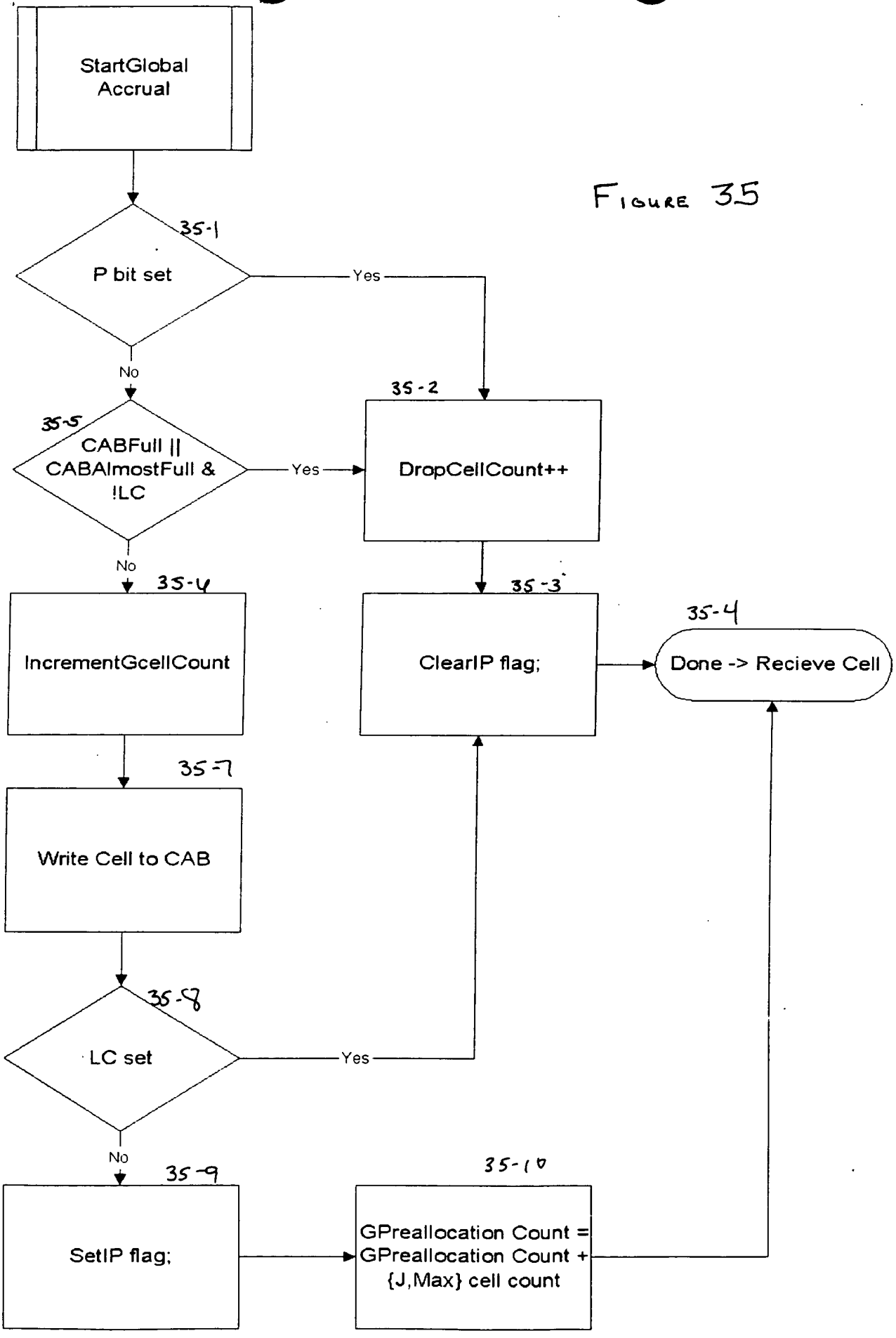

StartGlobal Accrual
35-1
P bit set
Yes
No
35-2
35-5
CABFull || CABAlmostFull & !LC
Yes
DropCellCount++
No
35-4
IncrementGcellCount
35-3
ClearIP flag;
35-4
Done -> Recieve Cell
35-7
Write Cell to CAB
35-8
LC set
Yes
No
35-9
SetIP flag;
35-10
GPreallocation Count = GPreallocation Count + {J,Max} cell count

FIGURE 35

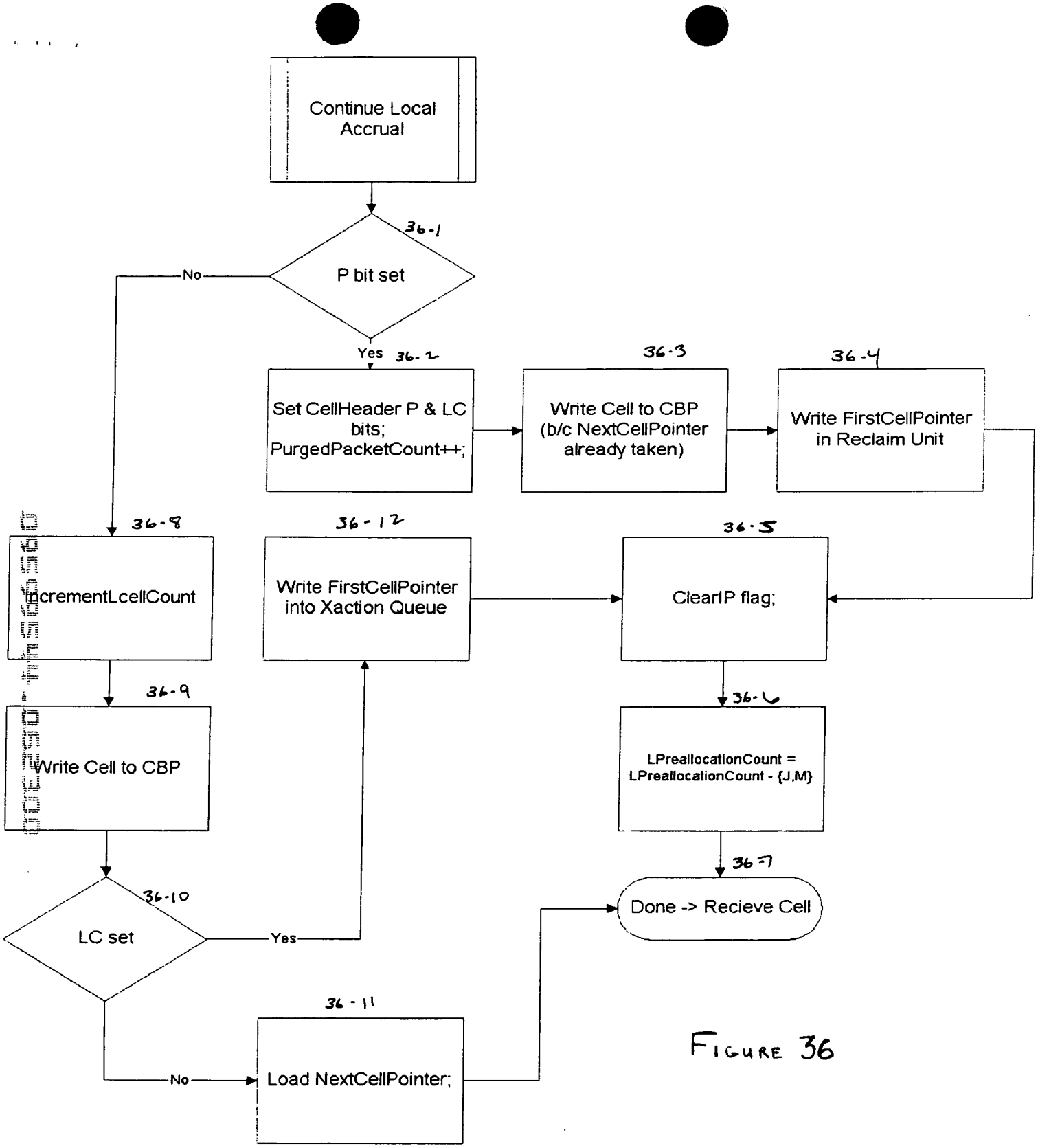

Continue Local Accrual
36-1
P bit set
No
Yes
36-2
Set CellHeader P & LC bits; PurgedPacketCount++;
36-3
Write Cell to CBP (b/c NextCellPointer already taken)
36-4
Write FirstCellPointer in Reclaim Unit
36-5
ClearIP flag;
36-6
LPreallocationCount = LPreallocationCount - {J,M}
36-7
Done -> Recieve Cell
36-8
IncrementLcellCount
36-9
Write Cell to CBP
36-10
LC set
Yes
36-11
Load NextCellPointer;
No
36-12
Write FirstCellPointer into Xaction Queue

FIGURE 36

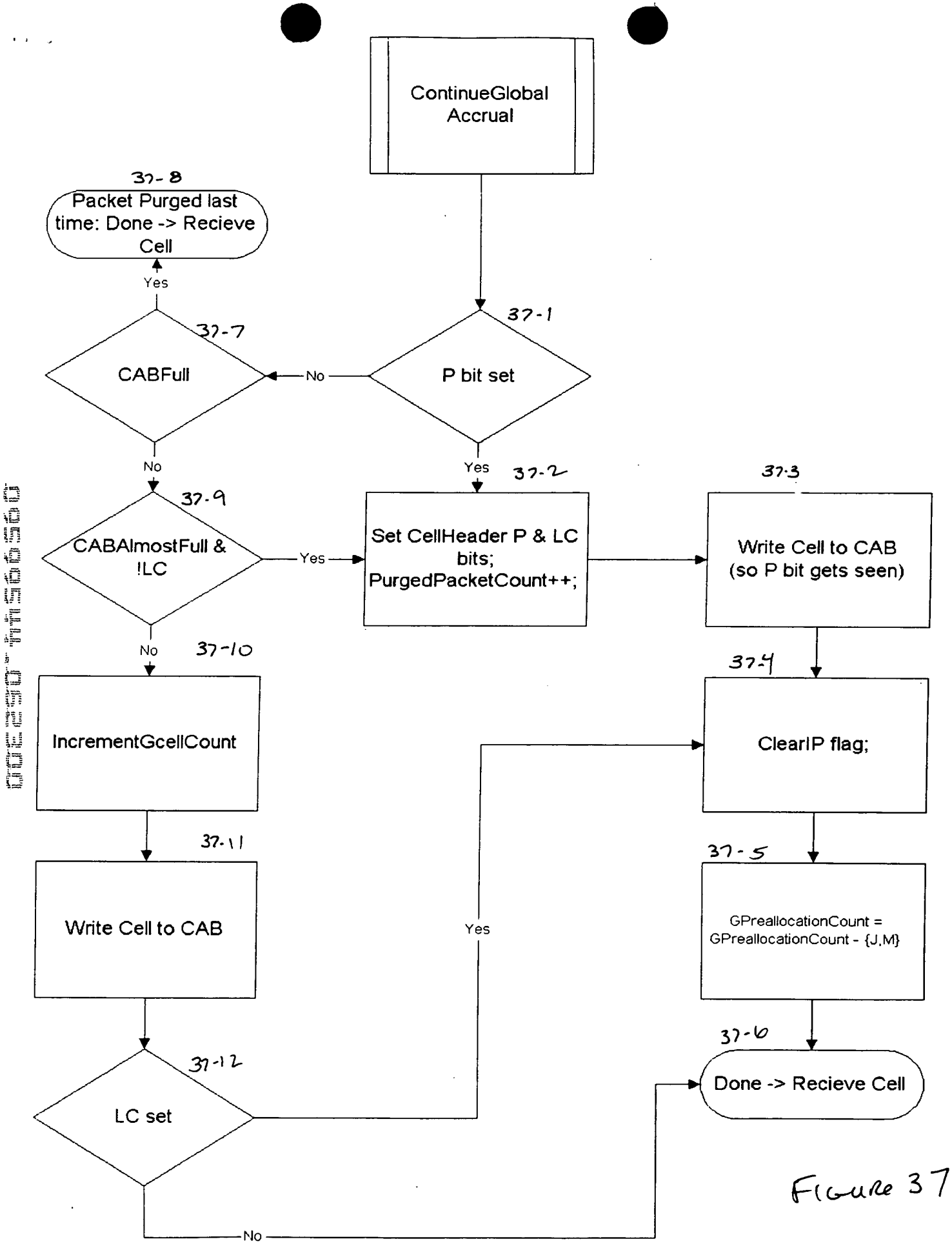

ContinueGlobal Accrual
37-8
Packet Purged last time: Done -> Recieve Cell
Yes
37-7
CABFull
No
37-1
P bit set
No
Yes
37-9
CABAlmostFull & !LC
Yes
37-2
Set CellHeader P & LC bits; PurgedPacketCount++;
37-3
Write Cell to CAB (so P bit gets seen)
No
37-10
IncrementGcellCount
37-4
ClearIP flag;
37-11
Write Cell to CAB
37-5
GPreallocationCount = GPreallocationCount - {J,M}
Yes
37-12
LC set
37-6
Done -> Recieve Cell
No

Figure 37

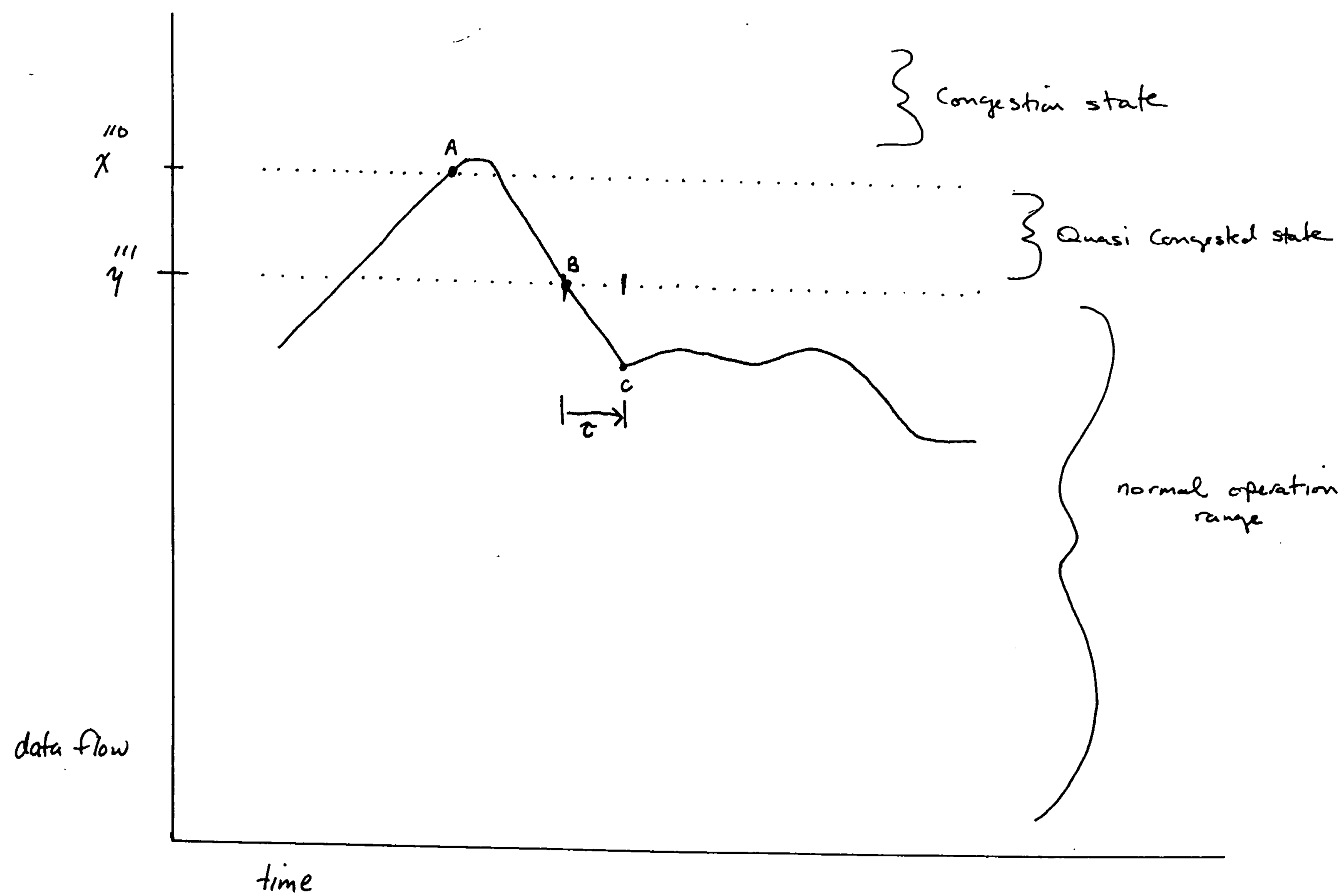

Congestion state
Quasi Congested state
normal operation range
A
B
C
τ
110
x
111
y
data flow
time

FIGURE 38

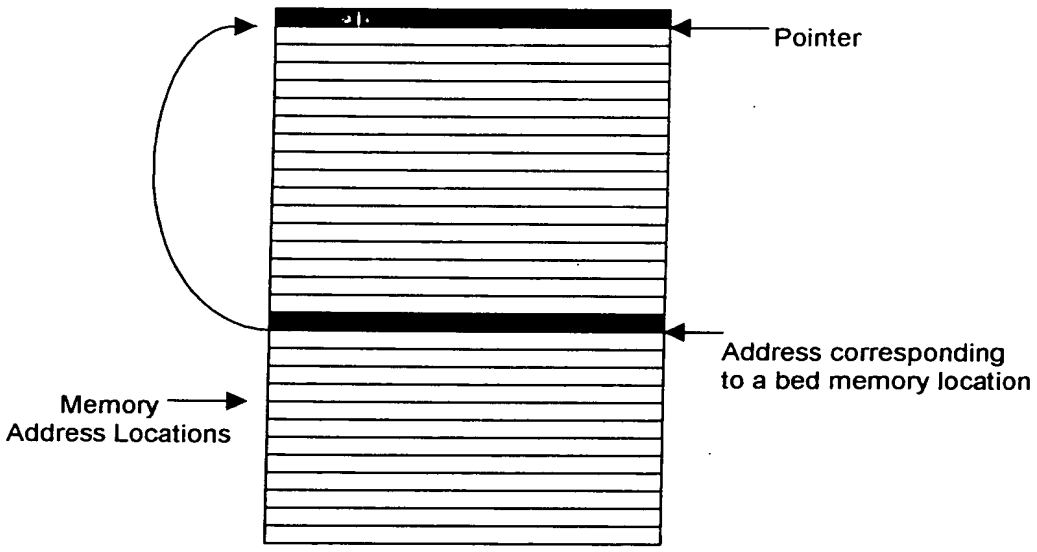

Pointer
Address corresponding
to a bed memory location
Memory
Address Locations

Figure 39